UNCLASSIFIED

1

AAI REPORT NO. R60011-00013

# AUTOMOTIVE TEST RIG FINAL DESIGN REPORT

JANUARY 1986

## VOLUME II – CONTROL SYSTEM

prepared for

U.S. NAVY DAVID TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER
CODE 1240, MARINE CORPS PROGRAMS OFFICE
BETHESDA, MARYLAND 20084

CONTRACT NO. N00167-84-C-0022

UNCLASSIFIED

89 2 16 206

UNCLASSIFIED

1

AAI REPORT NO. R60011-00013

# AUTOMOTIVE TEST RIG FINAL DESIGN REPORT

JANUARY 1986

## VOLUME II – CONTROL SYSTEM

~~DISTRIBUTION LIMITED: Further dissemination only as directed by the Commanding Officer, David Taylor Naval Ship Research and Development Center, ATTN: Code 1240, Marine Corps Programs Office, Bethesda, Maryland 20084.~~

prepared for

**U.S. NAVY DAVID TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER**
**CODE 1240, MARINE CORPS PROGRAMS OFFICE**
**BETHESDA, MARYLAND 20084**

CONTRACT NO. N00167-84-C-0022

APPROVED BY M.B. Hodges, III — SEQUENCE NO. B001

W/P LOG NO. M328 — DATA ITEM DI-S-4057

UNCLASSIFIED

89 2 16 206

SECURITY CLASSIFICATION OF THIS PAGE

# REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION IS UNLIMITED

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
R60011-00013

5. MONITORING ORGANIZATION REPORT NUMBER(S)
DTRC - SSID - CR - 5 - 89

6a. NAME OF PERFORMING ORGANIZATION
AAI CORPORATION

6b. OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION
DAVID TAYLOR RESEARCH CENTER

6c. ADDRESS *(City, State, and ZIP Code)*
P.O. BOX 6767
BALTIMORE, MD 21204

7b. ADDRESS *(City, State, and ZIP Code)*
CODE 1240
BETHESDA, MD 20084-5000

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
DAVID TAYLOR RESEARCH CENTER

8b. OFFICE SYMBOL *(If applicable)*
1240

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
N00167-84-C-0022

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

| PROGRAM ELEMENT NO. | PROJECT NO | TASK NO. | WORK UNIT ACCESSION NO. |
|---|---|---|---|
| 62543N | CF43455 |  | DN978568 |

11. TITLE *(Include Security Classification)*
AUTOMOTIVE TEST RIG FINAL DESIGN REPORT - VOL II CONTROL SYSTEM

12. PERSONAL AUTHOR(S)

13a. TYPE OF REPORT
FINAL

13b. TIME COVERED
FROM 1/84 TO 1/86

14. DATE OF REPORT *(Year, Month, Day)*
1986 January

15. PAGE COUNT
321

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

| FIELD | GROUP | SUB-GROUP |
|---|---|---|
|  |  |  |
|  |  |  |

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Tracked vehicle design, tracked amphibians, test bed, hydrostatic drive system, hydropneumatic suspension system, waterjets, micro-computer control, hydraulics land mobility, water mobility. (JES)

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

This report documents the design and fabrication of a tracked automotive test rig vehicle developed to investigate the feasibility of advanced automotivecomponents for potential use in future high water speed amphibians. Advanced components included in the design are a hydrostatic drive train for land and water propulsion, a hydropneumatic suspension system two speed final drives and micro computer control systems.

Volume I includes detailed descriptions of the various test bed major systems and components along with engineering analyses to support their design development. Characteristics of major test bed elements are defined along with overall test bed characteristics. Estimated land and water performance and weight and mass properties are also provided for the test bed.

Volume II provides the details of the hydrostatic drive train control system.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
Richard Swanek

22b. TELEPHONE *(Include Area Code)*
(301)227-1852

22c. OFFICE SYMBOL
1240

83 APR edition may be used until exhausted
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

# Table of Contents

Report No. R60011-00013
Volume II

Distribution Statement A is correct for this report.
Per Mr. Richard Swanek, DTRC/Code 1240

## List of Figures

Report No. R60011-00013
Volume II

# List of Tables

Report No. R60011-00013
Volume II

# 1.0 INTRODUCTION

This volume documents the technical aspects of the ATR Automotive control system and associated software. The information contained herein is an edited presentation of Southwest Research Institute's Final Report on the Control System Design for the Automotive Test Rig (ATR) Vehicle (SwRI-8284), December 1985. The report was prepared by Messrs. Gary L. Stecklein and Benjamin A. Treickel of the Department of Engine and Vehicle Research, Southwest Research Institute, San Antonio, Texas.

This volume addresses the control system design (hardware and software), auxiliary components (Terra Computer and Miltope Recorder), and transmission testing. Appendix A presents the detailed control system software listing and Appendix B provides details of the control circuits.

## 2.0 ATR CONTROL SYSTEM DESCRIPTION

Table 2.0-1 provides a listing of all of the operator input and vehicle feedback signals, and control signals used in the ATR vehicle. This table also indicates the pin number where these signals originate or terminate on the SC-1 computer. These devices can be categorized into the following major types:

- o Analog-In Signals
- o Frequency-In Signals
- o Digital Input Signals
- o Pulse Width Modulated (PWM) Output Signals
- o Digital Output Signals

These signals are discussed according to their category in the following section.

### 2.1 SC-1 COMPUTER INPUT SIGNALS

### 2.1.1 Operator Input Controls

The operator input signal devices consist of the steering mechanism, accelerator pedal, brake pedal, transmission selector, mode selector, and gear selector. Analog signals are provided by the steering, accelerator and brake devices. These signals are generated by potentiometers mounted in each of these devices. The input device displacement-to-signal generation-to-computer software integer value relationships are presented in Figures 2.1-1, 2.1-2 and 2.1-3 for these devices.

The mode selector, transmission selector, and gear selector provide digital input signals to the computer. Each device is mechanically interlocked so that only one input signal can be obtained. These signals are received by the SC-1 computer through a Parallel I/O card on card J18.

### 2.1.2 Speed Feedback Signals

Rotational speed feedback signals are obtained from the engine, the hydrostatic land drive motors, the sprockets, and the waterjet motors. These signals are used in the software to make power and performance calculations as well as during shifting operations where these speed signals are used to synchronize motor-to-clutch speeds.

These speeds are obtained through Frequency to Digital (F/D) converters. In principle these converters determine the number of internal

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer

| | |
|---|---|
| J7 | Pulse Width Modulated (PWM) Output Signal |
| P1 | Desired Low Clutch Valve Voltage (DLCVV) |
| P2 | Desired High Clutch Valve Voltage (DHCVV) |
| P3 | Desired Engine Speed (DES) |
| P4 | |
| P5 | Ground |
| J8 | Pulse Width Modulated (PWM) Output Signal |
| P1 | Port Forward Transmission Valve Voltage (PFTVV) |
| P2 | Starboard Forward Transmission Valve Voltage (SFTVV) |
| P3 | Port Aft Transmission Valve Voltage (PATVV) |
| P4 | Starboard Aft Transmission Valve Voltage (SATVV) |
| P5 | Ground |
| J9 | Digital to Analog Converter |
| P1 | Read Engine Speed (RES) |
| P4 | Read Port Motor Speed (RPMS) |
| P5 | Read Starboard Motor Speed (RSMS) |
| P9 | Computer Ground |
| P10 | Read Port Sprocket Speed (RPSS) |
| P11 | Read Starboard Sprocket Speed (RSSS) |
| J11 | Frequency to Digital (F/D) Converter |
| P11 | Ground |
| P16 | Input Actual Port Motor Speed (IAPMSP) |
| P17 | Input Actual Starboard Motor Speed (IASMSP) |
| P18 | Input Actual Port Sprocket Speed (IAPSSP) |
| P19 | Input Actual Starboard Sprocket Speed (IASSSP) |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer
(Continued)

| | |
|---|---|
| J12 | Frequency to Digital (F/D) Converter |
| P11 | Ground |
| P16 | Input Actual Port Waterjet Speed (IAPWSP) |
| P17 | Input Actual Starboard Waterjet Speed (IASWSP) |
| P18 | Input Actual Engine Speed (IAENSP) |
| P19 | |
| J13 | Analog to Digital (A/D Converter |
| P1 | Input Desired High Motor Speed (IDHMSP) |
| P2 | Input Desired High Motor Speed Ratio (IDHMSR) |
| P3 | Input Desired Turn Ratio (IDTRNR) |
| P4 | Input Port Bucket Angle (IAPBCN) |
| P5 | Input Starboard Bucket Angle (IASBCN) |
| P6 | |
| P7 | |
| P8 | |
| P9 | Ground |
| P10 | Ground |
| P11 | Ground |
| P12 | Ground |
| P13 | Ground |
| P14 | Ground |
| P15 | Ground |
| P16 | Ground |
| P17 | Computer Ground |
| J14 | Analog to Digital (A/D) Converter |
| P1 | Input Port Forward Motor Pressure (IPFMPR) |
| P2 | Input Starboard Forward Motor Pressure (ISFMPR) |
| P3 | Input Port Aft Motor Pressure (IPAMPR) |
| P4 | Input Starboard Aft Motor Pressure (ISAMPR) |
| P5 | Input Port Hydraulic Oil Temperature (IPHOTM) |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| P6 | Input Starboard Hydraulic Oil Temperature (ISHOTM) |
| P7 | Input Engine Coolant Temperature (IENCTM) |
| P8 | |
| P9 | Ground |
| P10 | Ground |
| P11 | Ground |
| P12 | Ground |
| P13 | Ground |
| P14 | Ground |
| P15 | Ground |
| P16 | Ground |
| P17 | Computer Ground |
| P18 | Computer Ground |
| J15 | Analog to Digital (Aux Analog In) (Required Separate Connector on SC-1 Computer Enclosure) |
| P1 | |
| P2 | |
| P3 | |
| P4 | |
| P5 | |
| P6 | |
| P7 | |
| P8 | |
| P9 | Ground |
| P10 | Ground |
| P11 | Ground |
| P12 | Ground |
| P13 | Ground |
| P14 | Ground |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| P15 | Ground |
| P16 | Ground |
| P17 | Computer Ground |
| P18 | Computer Ground |
| | |
| J16 | Parallel I/O (Switch Closures to Ground) |
| P1 | Clamp (+24) |
| P18 | Desired Suspension Up Valve Status |
| P17 | Desired Suspension Down Valve Status |
| P16 | Desired Secondary Cooling Fan 4 gpm Valve Status |
| P15 | Desired Secondary Cooling Fan 8 gpm Valve Status |
| P14 | Desired Electric Bilge Pump Activation |
| P13 | |
| P12 | Desired Starboard Bucket Counterclockwise Valve Status |
| P11 | Desired Starboard Bucket Clockwise Valve Status |
| P10 | Clamp (+24) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |
| P4 | |
| P3 | |
| P2 | |
| P20 | Clamp (+24) |
| P28 | Low Brake and Clutch Lube Pressure Switch Status |
| P27 | Low Brake Release Pressure Switch Status |
| P26 | Low Brake Supply Pressure Switch Status |
| P25 | Low Port Charge Pump Pressure Switch Status |
| P24 | Low Starboard Charge Pump Pressure Switch Status |
| P23 | Hydraulic Filter By-Pass Switch Status |
| P22 | Port 1 Suspension Component Switch Status |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| P21 | Port 2 Suspension Component Switch Status |
| P19 | Port 3 Suspension Component Switch Status |
| P29 | Cathode Signal Return |
| P36 | Port 4 Suspension Component Switch Status |
| P35 | Port 5 Suspension Component Switch Status |
| P34 | Low Suspension System Supply Pressure |
| P33 | |
| P32 | |
| P31 | |
| P30 | |
| P37 | Signal Return for Computer |
| J17 | Parallel I/O (Switch Closure to Ground) |
| P1 | Clamp (+24) |
| P18 | |
| P17 | Desired Primary Cooling Fan on Valve Status |
| P16 | Desired Primary Cooling Fan Stop Valve Status |
| P15 | Desired Secondary Grill Open Valve Status |
| P14 | |
| P13 | Desired Seawater Pump Activation |
| P12 | Desired Waterjet By-Pass Valve Status |
| P11 | Desired Waterjet No By-Pas Valve Status |
| P10 | Clamp (+24) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |
| P4 | |
| P3 | |
| P2 | |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| P20 | (Clamp (+24) |
| P28 | Starboard 1 Suspension Component Switch Status |
| P27 | Starboard 2 Suspension Component Switch Status |
| P26 | Starboard 3 Suspension Component Switch Status |
| P25 | Starboard 4 Suspension Component Switch Status |
| P24 | Starboard 5 Suspension Component Switch Status |
| P19 | |
| P29 | Cathode Signal Return |
| P36 | Low Vehicle System Voltage Switch Status |
| P35 | Low Control System Voltage Switch Status |
| P34 | High Vehicle System Voltage Switch Status |
| P33 | High Control System Voltage Switch Status |
| P32 | Primary Grill Closed Switch Status |
| P30 | |
| P37 | Signal Return for Computer |
| J18 | Parallel I/O (Positive Voltage to Card) |
| P1 | Clamp (+24) |
| P18 | Desired Motor No By-Pass Valve Status |
| P17 | Desired Port Bucket Counterclockwise Valve Status |
| P16 | Desired Port Bucket Clockwise Valve Status |
| P15 | Desired Primary Grill Closure Valve Status |
| P14 | Desired Primary Grill Open Valve Status |
| P13 | Desired Hydraulic Bilge Pump Valve Activation |
| P12 | Desired Hydraulic By-Pass Valve Status |
| P11 | Desired Motor By-Pass Valve Status |
| P10 | Clamp (+24) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| P4 | |
| P3 | |
| P2 | |
| P20 | Clamp (+24) |
| P28 | Aft Electric Bilge Pump Activation Status |
| P27 | Seawater Pump Activation Status |
| P26 | Hydraulic Bilge Pump Valve Activation Status |
| P25 | Forward Electric Bilge Pump Activation Status |
| P24 | Desired High Gear Selector Switch Status |
| P23 | Desired Low Gear Selector Switch Status |
| P22 | Desired Seaborne Mode Selector Switch Status |
| P21 | Desired Transition Mode Selector Switch Status |
| P19 | Desired Landborne Mode Selector Switch Status |
| P29 | Cathode Signal Return |
| P36 | Desired Drive Selector Switch Status |
| P35 | Desired Neutral Selector Switch Status |
| P34 | Desired Reverse Selector Switch Status |
| P33 | Desired Park Selector Switch Status |
| P32 | Fire Sensed Switch Status |
| P31 | Low Scavenge Pump Pressure Switch Status |
| P30 | |
| P37 | Signal Return for Computer |
| J21 | Not Used |
| J22 | |
| P | A +24 |
| | B Pwr Rtn |
| | C Ground |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| J23 | Not Used |
| J24 | Not Used |
| J25 | |
| P1 | +24 |
| P2 | High Level Output (Forward Electric Bilge Pump Activation) |
| P3 | Low Level Input |
| P4 | +24 |
| P5 | High Level Output (Aft Electric Bilge Pump Activation) |
| P6 | Low Level Input |
| P7 | +24 |
| P8 | High Level Output (Suspension Down Valve Activation) |
| P9 | Low Level Input |
| P10 | +24 |
| P11 | High Level Output (Suspension Up Valve Activation) |
| P12 | Low Level Input |
| P13 | +24 |
| P14 | High Level Output |
| P15 | Low Level Input |
| P16 | +24 |
| P17 | High Level Output |
| P18 | Low Level Input |
| P19 | +24 |
| P20 | High Level Output |
| P21 | Low Level Input |
| P22 | +24 |
| P23 | High Level Output |
| P24 | Low Level Input |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

J26

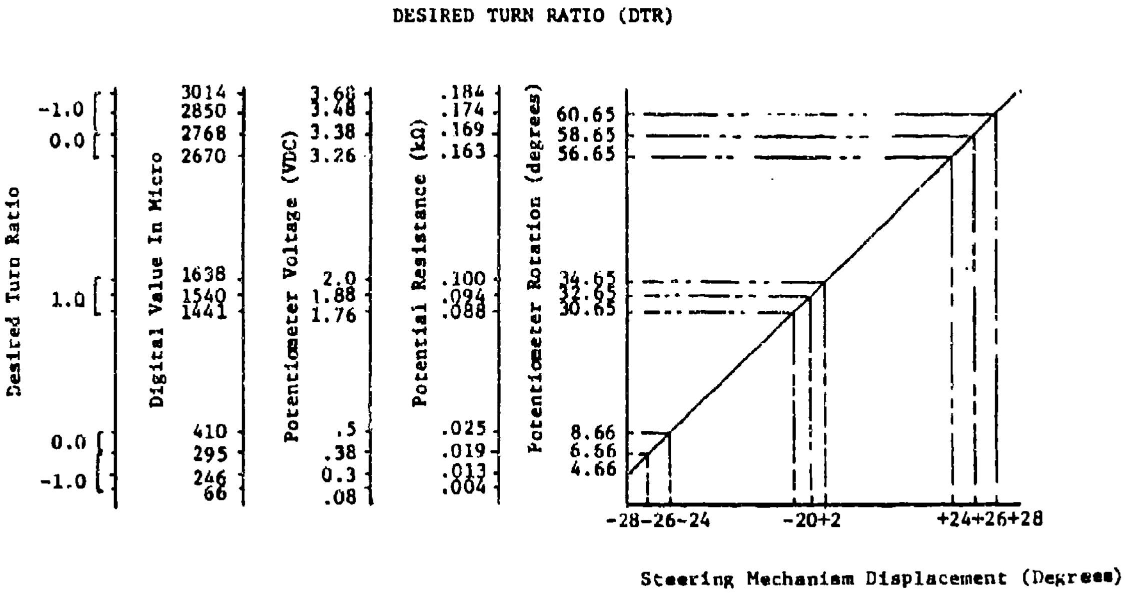
| | |
|---|---|
| P1 | +24 |
| P2 | PWM Output (Desired Low Clutch Valve Voltage (DCLVV)) |
| P3 | PWM Input from J7-1 |
| P4 | +24 |
| P5 | PWM Output (Desired High Clutch Valve Voltage (DCLVV)) |
| P6 | PWM Input from J7-2 |
| P7 | +24 |
| P8 | PWM Output (Output Desired Engine Speed (DES)) |
| P9 | PWM Input from J7-3 |
| P10 | +24 |
| P11 | PWM Output |
| P12 | PWM Input from J7-4 |
| P13 | +24 |
| P14 | PWM Output (Desired Port Forward Transmission Valve Voltage (DPFTVV)) |
| P15 | PWM Input from J8-1 |
| P16 | +24 |
| P17 | PWM Output (Desired Starboard Forward Transmission Valve Voltage (DSFTVV) |
| P18 | PWM Input from J8-2 |
| P19 | +24 |
| P20 | PWM Output (Desired Port Aft Transmission Valve Voltage (DPATVV)) |
| P21 | PWM Input from J8-3 |
| P22 | +24 |
| P23 | PWM Output (Desired Starboard Aft Transmission Valve Voltage (DSATVV)) |
| P24 | PWM Input from J8-4 |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | |
|---|---|
| J27 | |
| P1 | |
| P2 | |
| P3 | |
| P4 | |
| P5 | |
| P6 | |
| P7 | |
| P8 | |
| P9 | |
| P10 | +5V |
| P11 | +5V |
| P12 | Reset Switch Lead (Spring Loaded Normally Open) |
| P13 | Reset Switch Lead |
| P14 | |
| P15 | |
| P16 | |
| P17 | |
| P18 | |
| P19 | |
| P20 | |
| P21 | |
| P22 | |
| P23 | |
| P24 | |
| P25 | |
| J28 | RS 232 Interface |
| P1 | |
| P2 | Xmit |
| P3 | Rcv |
| P4 | |

Table 2.0-1. Input/Output Signal Listing by Pin Number for the SC-1 Computer (Continued)

| | | |
|---|---|---|
| P5 | | |
| P6 | | |
| P7 | Sig Ret | |
| P8 | | |
| P9 | | |
| P10 | | |
| P11 | | |
| P12 | | |
| P13 | | |
| P14 | | |
| P15 | | |
| P16 | | |
| P17 | | |
| P18 | | |
| P19 | | |
| P20 | | |
| P21 | | |
| P22 | | |
| P23 | +15V | |
| P24 | Pwr Rtn | |
| P25 | -15V | |
| J29 | | |
| P | A | +24V |
| | B | Pwr Rtn |
| | C | Ground |



Pot Rotation: degrees = Steering Mechanism Rotation: degrees +32.48

$$\text{Pot Resistance: k}\Omega = \left(\frac{\text{Pot Rotation: degrees}}{348}\right)(1\text{k}\Omega)$$

Pot Voltage: VDC = (20mA) (Pot Resistance: kΩ)

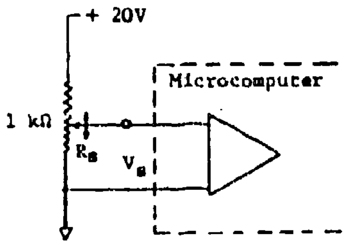
$$\text{Micro Integer Value} = \frac{\text{Pot Voltage: VDC}}{5} \quad 4095$$

For Left Turns

$$\text{Desired Turn Ratio} = (.00097) \begin{array}{l}\text{Micro}\\ \text{Integer}\\ \text{Value}\end{array} -.04$$

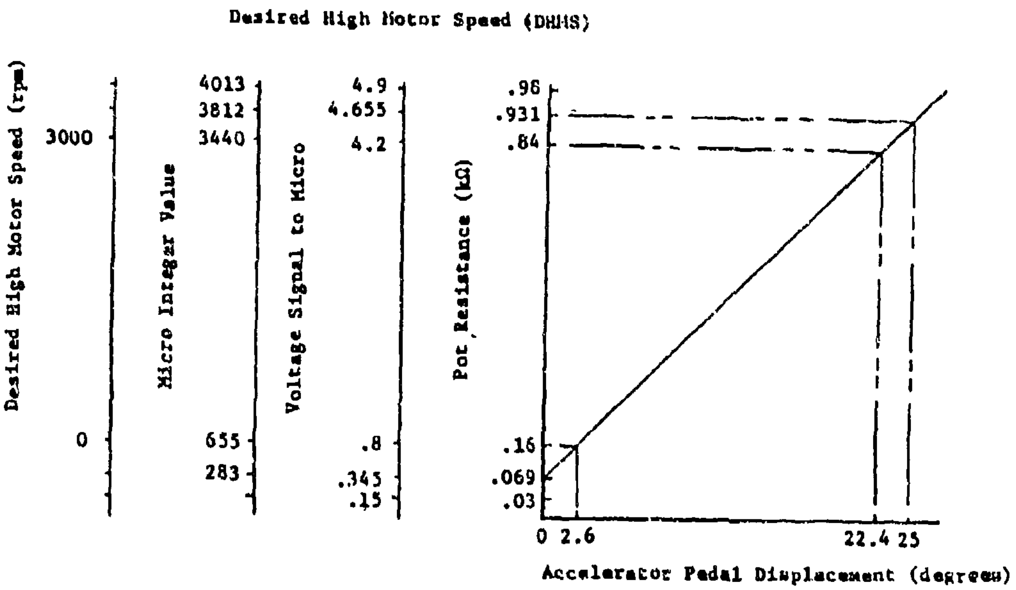
For Right Turns

$$\text{Desired Turn Ratio} = (-.00097) \begin{array}{l}\text{Micro}\\ \text{Integer}\\ \text{Value}\end{array} + 2.587$$



$$i = \frac{20V}{1\ \text{k}\Omega} = 20\ \text{mA}$$

Figure 2.1-1 Steering Mechanism Displacement Signal Relationships



Pot Resistance 1kΩ = (Accelerator Displacement: degrees) $\left(\frac{300}{25}\right)\left(\frac{1k\Omega}{348}\right) + .069$

Pot Resistance: kΩ = (Accelerator Displacement: degrees)(.0345) + .069

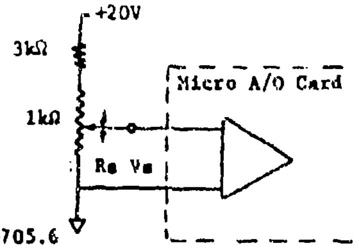
Voltage Signal To Micro: VDC = (Pot Resistance: kΩ)(5mA)

Integer Value In Micro = $\frac{\text{Voltage Signal to Micro}}{\text{5VDC}}$ (4095)

$$\begin{array}{r} 3000 = 3440x + y \\ 0 = 655x + y \\ \hline 3000 = 2785x \end{array}$$

$x = 1.077$
$y = -705.6$

Desired High Motor Speed: rpm = 1.077 (Micro Integer Value) -705.6



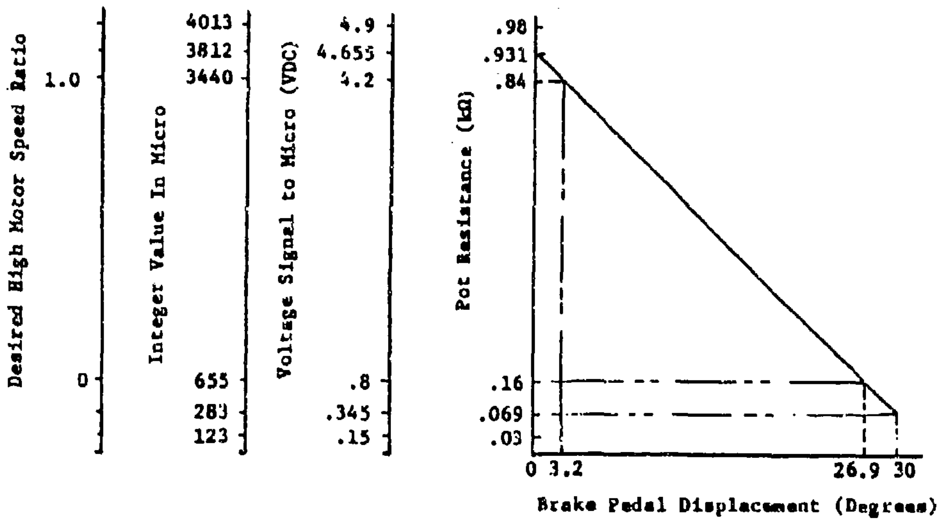
$$5VDC = (1k\Omega)(i)$$
$$i = 5mA$$

$$R_{xmin} = \frac{1k\Omega - \frac{300}{348}\ 1k\Omega}{2} = .069k\Omega$$

$$R_{xmax} = 1k\Omega - .069\ k\Omega = .931\Omega$$

Figure 2.1-2 Accelerator Pedal Displacement Signal Relationships

Desired High Motor Speed Ratio (DHMSR)



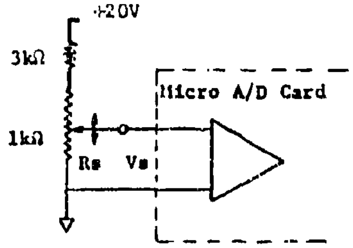
Pot Resistance: kΩ = (Brake Pedal Displacement: degrees)$\left(\frac{300}{30}\right)\left(\frac{1k\Omega}{348}\right)$+ .069

Pot Resistance: kΩ = (Brake Pedal Displacement: degrees)(.0287) + .069

Voltage Signal to Micro: VDC = (Pot Resistance: kΩ)(5mA)

$$\begin{array}{r} 1.0 = 3440x + y \\ \underline{0 = 655x + y} \\ 1.0 = 2785x \end{array}$$

$x = .0003591$
$y = .2352$



5VDC = (1kΩ)(i)
i = 5mA

Desired High Motor Speed Ratio = .0003591 (Micro Integer Value) - .2352

Figure 2.1-3 Brake Pedal Displacement
Signal Relationships

clock counts (from an internal clock that pulses at a rate of 5.1177 x $10^5$ counts/sec) that occur between the passage of gear teeth. This provides a direct period of measurement of the gear tooth passage which can be directly converted to rotational speed. Figures 2.1-4, -5, -6 and -7 present the relationships between speed and computer-integer value for the engine, land drive motors, sprockets and waterjet motors, respectively.

### 2.1.3 Waterjet Bucket Angle Signals

While in the transition or seaborne mode of operation, the waterjet buckets are rotated to generate turning forces. The steering mechanism displacment dictates the desired waterjet bucket rotation angle. Each waterjet bucket position is individually monitored to determine if it should be rotated clockwise, counterclockwise, or if its position is satisfactory. Feedback signals or the actual wacerjet bucket position are used for this analysis. Figures 2.1-8 and -9 provide the port starboard waterjet bucket angle-to-computer software integer value relationships.

### 2.1.4 Hydrostatic Transmission Pressure Signals

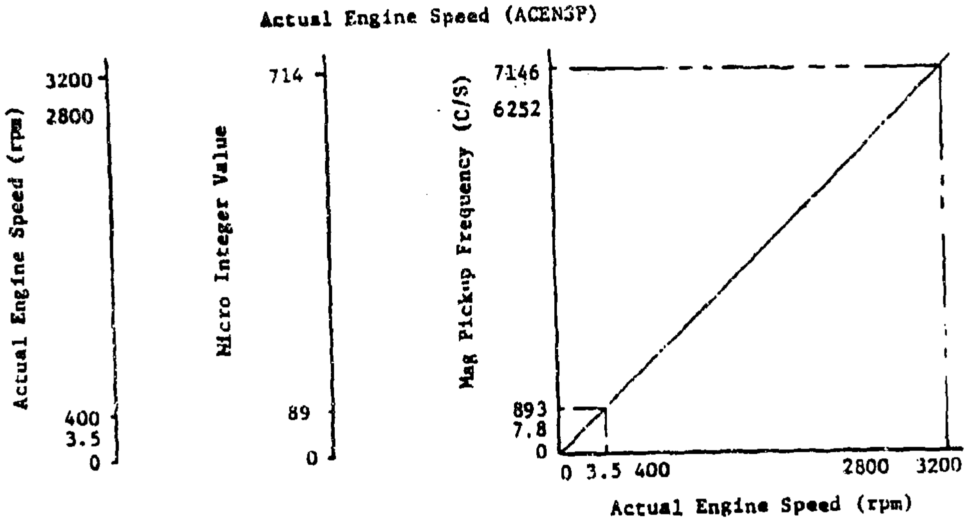
Hydrostatic transmission feedback signals are used to determine the torques on the hydrostatic motors and power requirements during transition and seaborne operation. Current loop pressure transmitters are used for this application which have a 10,000 psi working pressure rating. The pressure-to-computer integer relationship for these devices is presented in Figure 2.1-10.

### 2.1.5 Working Fluid Temperature Signals

During vehicle operation, the port and starbord hydraulic oil temperatures and engine coolant temperature are continuously monitored to determine when the various cooling fans should be activated and when over-temperature conditions exist. Resistance thermometer devices (RTDs) are used in this application in combination with current loop transmitters which convert the low level RTD voltage signal to a current signal. Figure 2.1-11 relates the RTD output current signal-to-computer software integer value relationships.

### 2.1.6 Digital Input Signals

The digital input signals listed in Table 2.0-1 consist of two major types which can be thought of as either switch signal closing to gr or a switch signal closing to a voltage source. The input signals associ



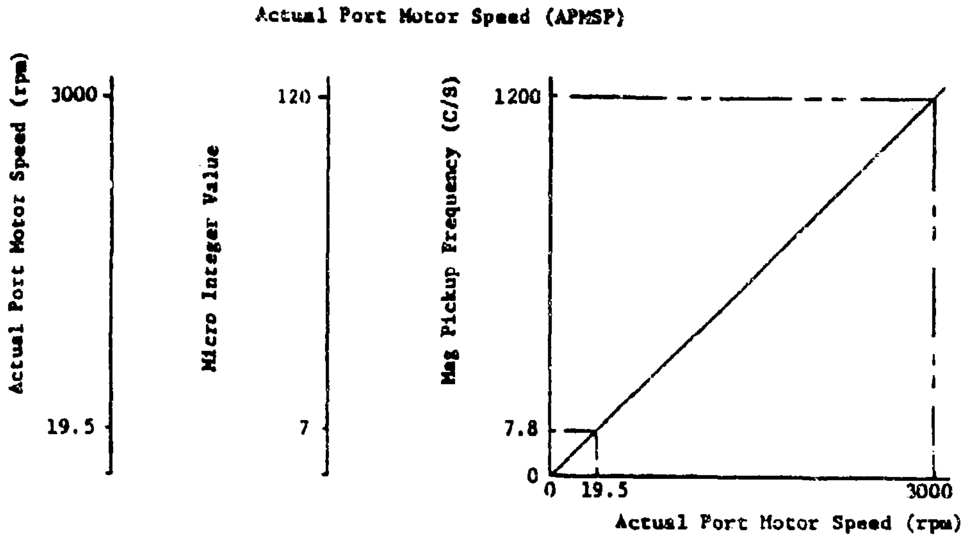
Mag Pickup Frequency: C/S = (Actual Engine Speed: rpm)$\left(\frac{134\ \text{C/rev}}{60\ \text{sec/min}}\right)$

Mag Pickup Frequency: C/S = (Actual Engine Speed: rpm) (2.233 C/S//rpm)

Micro Integer Value = (Mag Pickup Frequency: C/S)(10.0)

Actual Engine Speed: rpm = $\frac{\text{(Micro Integer Value)(600)}}{134}$

Figure 2.1-4 Engine Speed Signal Relationships



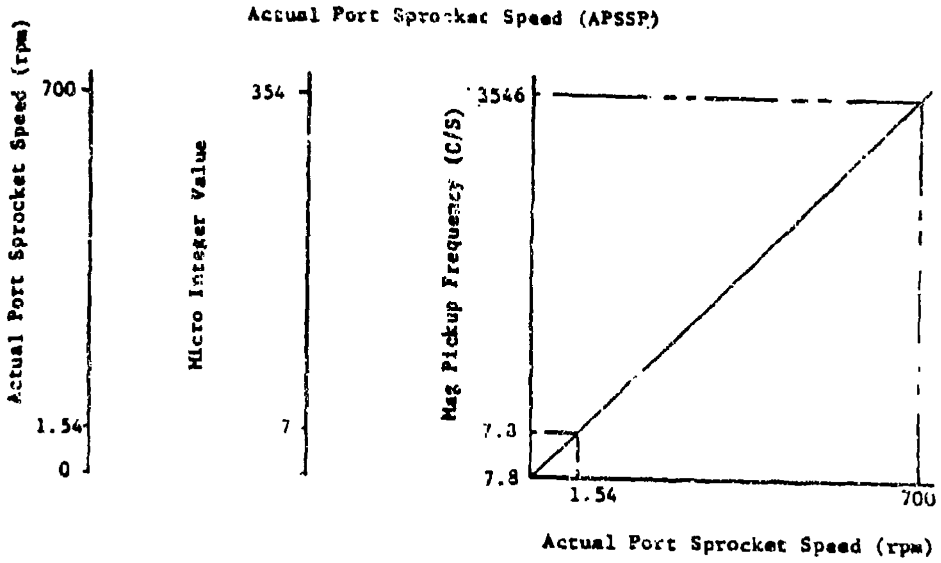
Mag Pickup Frequency: $C/S = (\text{Actual Port Motor Speed: rpm})\left(\frac{24\ \text{C/rev}}{60\ \text{sec/min}}\right)$

Mag Pickup Frequency: $C/S = (\text{Actual Port Motor Speed: rpm})\ (.4\ \text{C/S//rpm})$

Micro Integer Value = (Mag Pickup Frequency: C/S)(10.0)

Actual Port Motor Speed: $\text{rpm} = \frac{(\text{Micro Integer Value})(600)}{24}$

Figure 2.1-5 Land Drive Motor Speed Signal Relationships



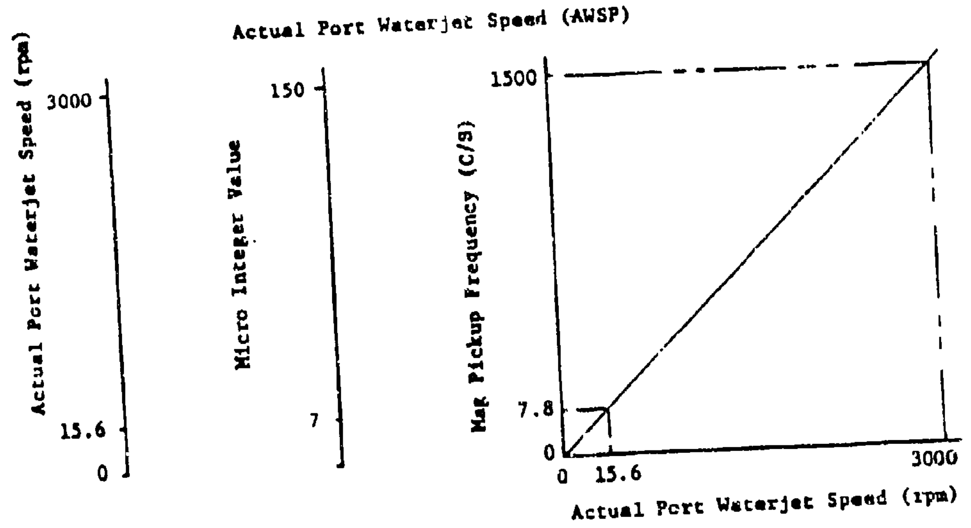
Mag Pickup Frequency: C/S = (Actual Port Sprocket Speed: rpm)$\left(\frac{(2.338)(130\ \text{C/rev})}{60\ \text{sec/min}}\right)$

Mag Pickup Frequency: C/S = (Actual Port Sprocket Speed: rpm) (5.066 C/S//rpm)

Micro Integer Value = (Mag Pickup Frequency: C/S)(10.0)

Actual Port Sprocket Speed: rpm = $\frac{(\text{Micro Integer Value})(600)}{130}$

Figure 2.1-6 Sprocket Speed Signal Relationships



Mag Pickup Frequency: C/S = (Actual Port Waterjet Speed: rpm)$\left(\frac{\text{30 C/rev}}{\text{60 sec/min}}\right)$

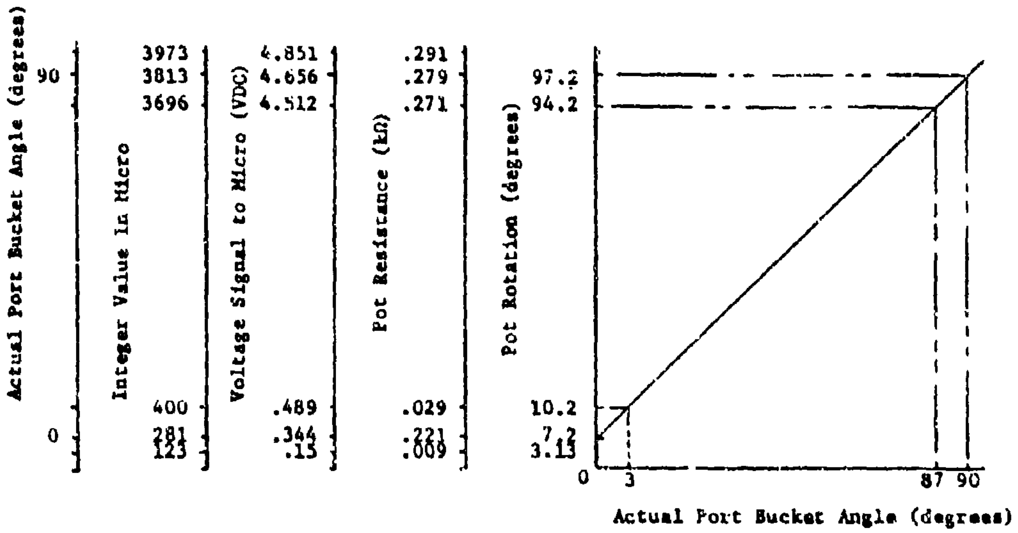
Mag Pickup Frequency: C/S = (Actual Port Waterjet Speed: rpm) (.5 C/S//rpm)

Micro Integer Value = (M ; Pickup Frequency: C/S)(10.0)

Actual Port Waterjet Speed: rpm = $\frac{\text{(Micro Integer Value)(600)}}{30}$

Figure 2.1-7 Waterjet Speed Signal Relationships

Actual Port Bucket Angle (APBCAP)

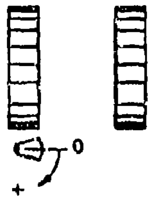


Port Rotation degrees = (Bucket Rotation: degrees) + 7.2

Pot Resistance: kΩ = $\left(\frac{\text{Pot Rotation}}{348}\right)$ 1kΩ

Voltage Signal to Micro: VDC = (Pot Resistance: kΩ)(16.67mA)

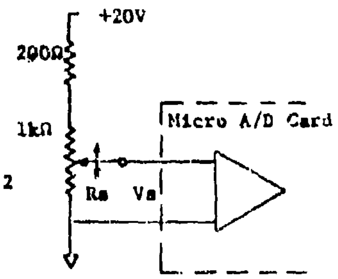
Integer Value in Micro = $\left(\frac{\text{Pot Voltage: VDC}}{\text{5VDC}}\right)$ 4095



$$\begin{aligned} 90 &= 3813x + y \\ 0 &= 282x + y \\ \hline 90 &= 3513x \end{aligned}$$

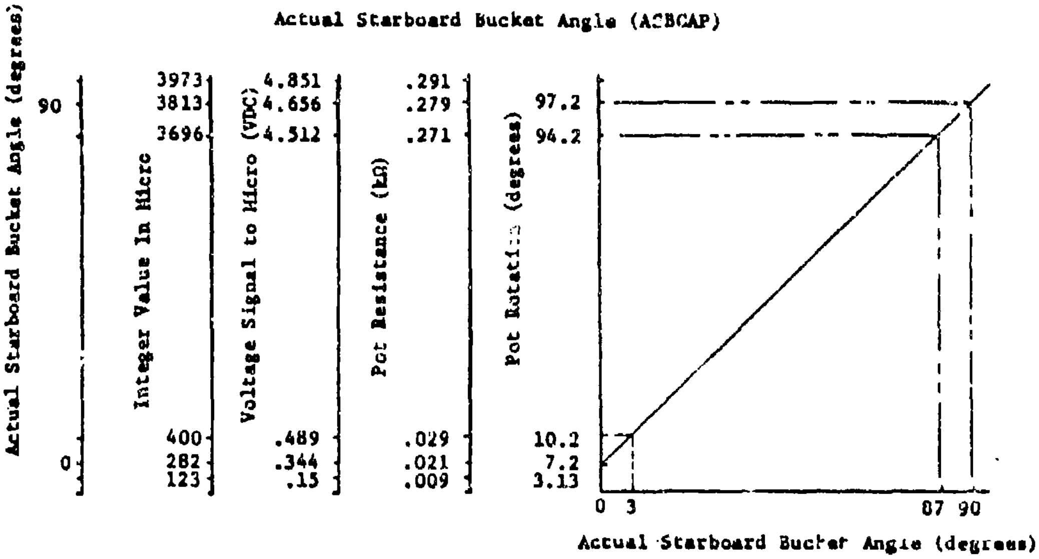
x = .0255
y = -7.2

Actual Port Bucket Angle: degrees = (.0255)(Integer Value)-7.2

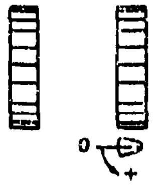


20VDC = (1.20 kΩ)(i)
i = 16.67mA

Figure 2.1-8 Port B :ket Angle Signal Relationships



Pot Rotation: degrees = (Bucket Rotation: degrees) + 7.2



Pot Resistance: kΩ = $\left(\frac{\text{Pot Rotation}}{348}\right)$ 1kΩ

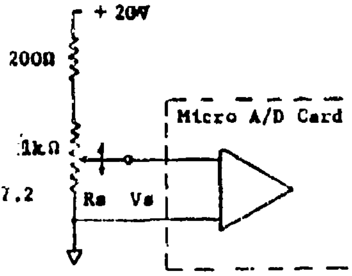
Voltage Signal to Micro: VDC = (Pot Resistance: kΩ)(16.67mA)

Integer Value in Micro = $\left(\frac{\text{Pot Voltage: VDC}}{\text{5VDC}}\right)$ 4095

$$\begin{aligned} 90 &= 3813x + y \\ 0 &= 282x + y \\ \hline 90 &= 3513x \end{aligned}$$

x = .0255
y = -7.2

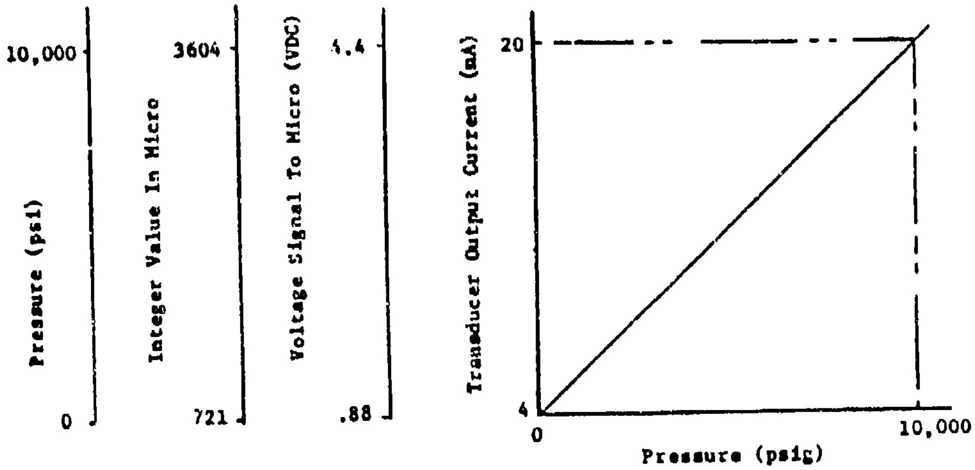
Actual Starboard Bucket Angle: degrees = (.0255)(Integer Value) -7.2



20VDC = (1.20 kΩ)(i)
i = 16.67mA

Figure 2.1-9 Starboard Bucket Angle Signal Relationships

Port Aft Motor Pressure



$$\begin{array}{l} 20 = 10000x + y \\ \underline{4 \;\; = 0x + y} \\ 16 = 10000x \end{array}$$

$x = .0016 \qquad y = 4$

Transducer Output Current imA = (.0016)(Pressure ipsig) + 4

Voltage Signal To Micro: VDC = (Transducer Output Current: mA)(.22kΩ)

$$\text{Micro Integer Value} = \frac{\text{Voltage Signal To Micro}}{\text{5VDC}} \; 4095$$

$$\begin{array}{l} 10000 = 3604x + y \\ \underline{0 \qquad = 721x + y} \\ 10000 = 2883x \end{array}$$

$x = 3.469 \qquad y = -2501$

Pressure psig = (3.469)(Micro Integer Value) -2501

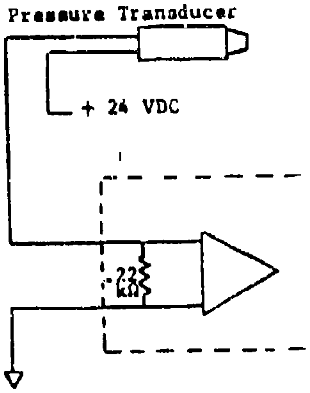
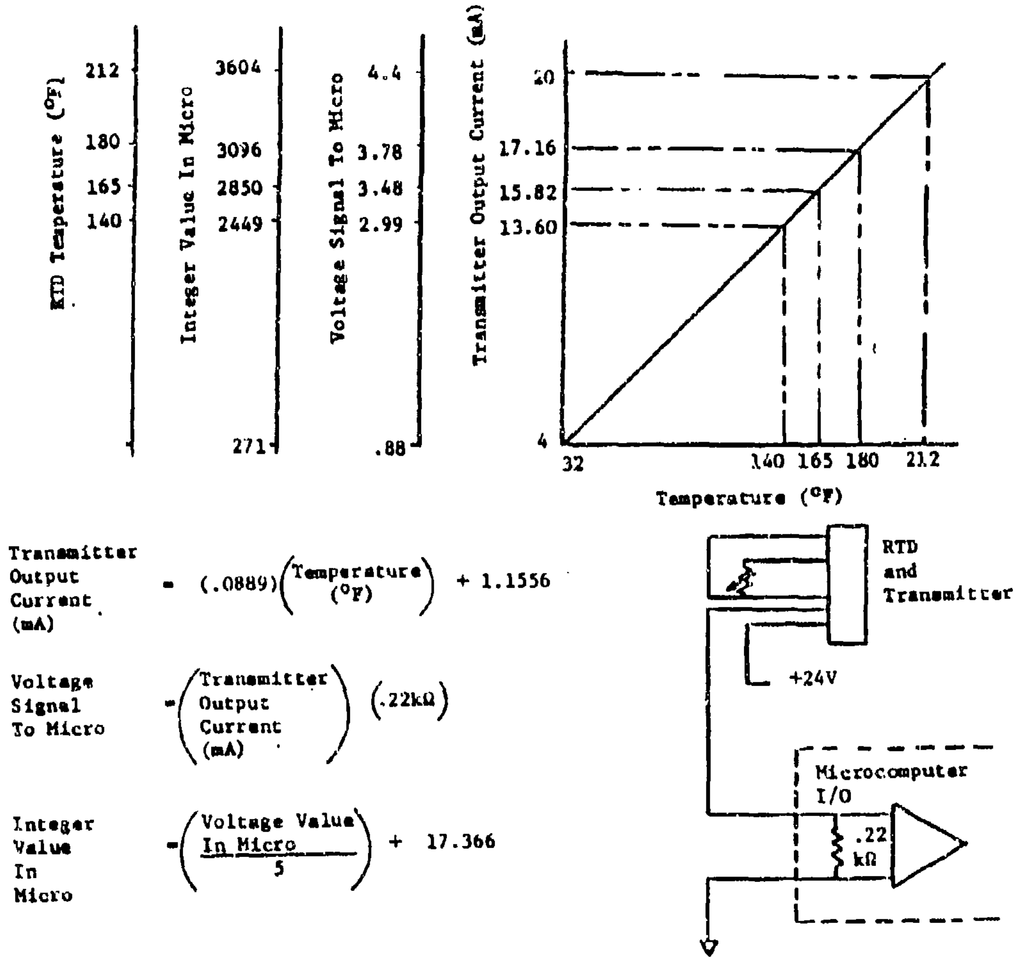


Figure 2.1-10 Hydrostatic Pressure Signal Relationships



Transmitter Output Current (mA) $= (.0889)\left(\begin{matrix}\text{Temperature}\\ (^\circ\text{F})\end{matrix}\right) + 1.1556$

Voltage Signal To Micro $= \left(\begin{matrix}\text{Transmitter}\\ \text{Output}\\ \text{Current}\\ (\text{mA})\end{matrix}\right)(.22\text{k}\Omega)$

Integer Value In Micro $= \left(\dfrac{\text{Voltage Value In Micro}}{5}\right) + 17.366$

Figure 2.1-11 Temperature Signal Relationships

with the J16 and J17 parallel I/O cards are switch closures to ground, while the input signals associated with J18 are switch closures to a voltage supply. For switch closures to ground, the computer parallel I/O cards (Digital In/Digital Out) cards must provide the power necessary to provide a signal when the switch is closed to ground indicating an error condition. In this case the parallel I/O card acts as a current source. When the switch is closed to a voltage source indicating an error condition, however, the parallel I/O card must act as a current sink. Figures 2.1-12 and -13 provide electrical schematics of these two signal types and their connection to the parallel I/O card.

## 2.2 SC-1 Computer Output Signals

### 2.2.1 Digital Output Signals

The digital output signals generated by the computer are presented in Table 2.0-1 on I/O cards J16-J18. For all of these signals the parallel I/O card acts like a current sink. Since all of the digital output signals are driving high power level inductive devices, a relay circuit is provided to allow the low level computer signal to energize the motor or valve solenoid inductive devices. A typical schematic for these circuits is shown in Figure 2.2-1.

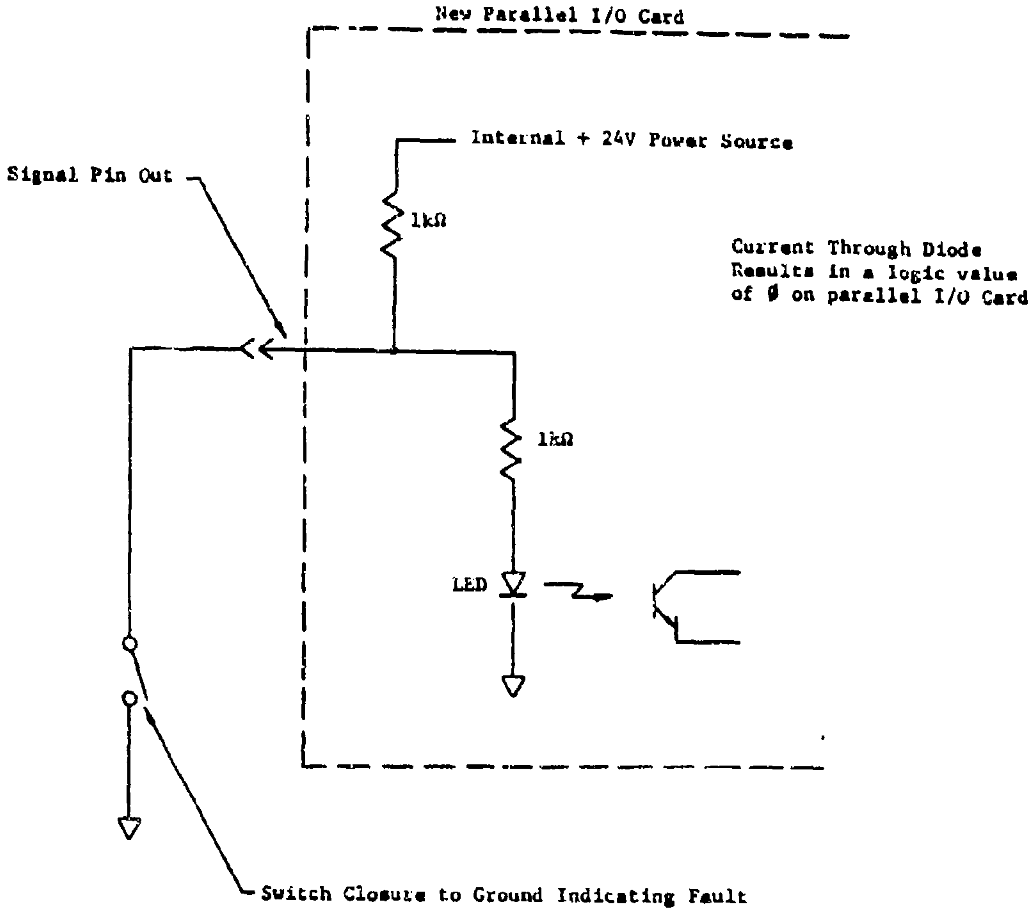
### 2.2.2 Pulse Width Modulated (PWM) Output Signals

PWM signals are provided to drive the transmission control valves, clutch control valves, and engine governor controller. These s gnals are generated at a low voltage value, +5V, in the microcomputer I/O chassis and are upgraded to the +24V signal level and 1.0 amp power level through the use of bipolar junction transistor driver circuits also housed within the I/O chassis.

These sig als are all operated with a fixed frequency of 300 hz and duty cycles variable between 0.0 and 100.0 percent. This yields average voltage levels variable between 0.0 and 24.0 volts.

## 2.3 SC-1 Computer Description

An outline drawing of the computer illustrating the location of the various I/O functions (F/A, A/D, D/A, PWM, an Parallel I/O) is presented as Figure 2.3-1. This figure can be compared to the signal listing provided as Table 2.0-1 to obtain a visual indication of where the input and output signals will be received and transmitted by the computer.


New Parallel I/O Card
Internal + 24V Power Source
Signal Pin Out
1kΩ
Current Through Diode Results in a logic value of Ø on parallel I/O Card
1kΩ
LED
Switch Closure to Ground Indicating Fault

Partial Signal Schematic for Inputs To Two New Parallel I/O Card For ATR Project

Figure 2.1-12 Digital Input Signal Schematic as a Switch Closure to Ground

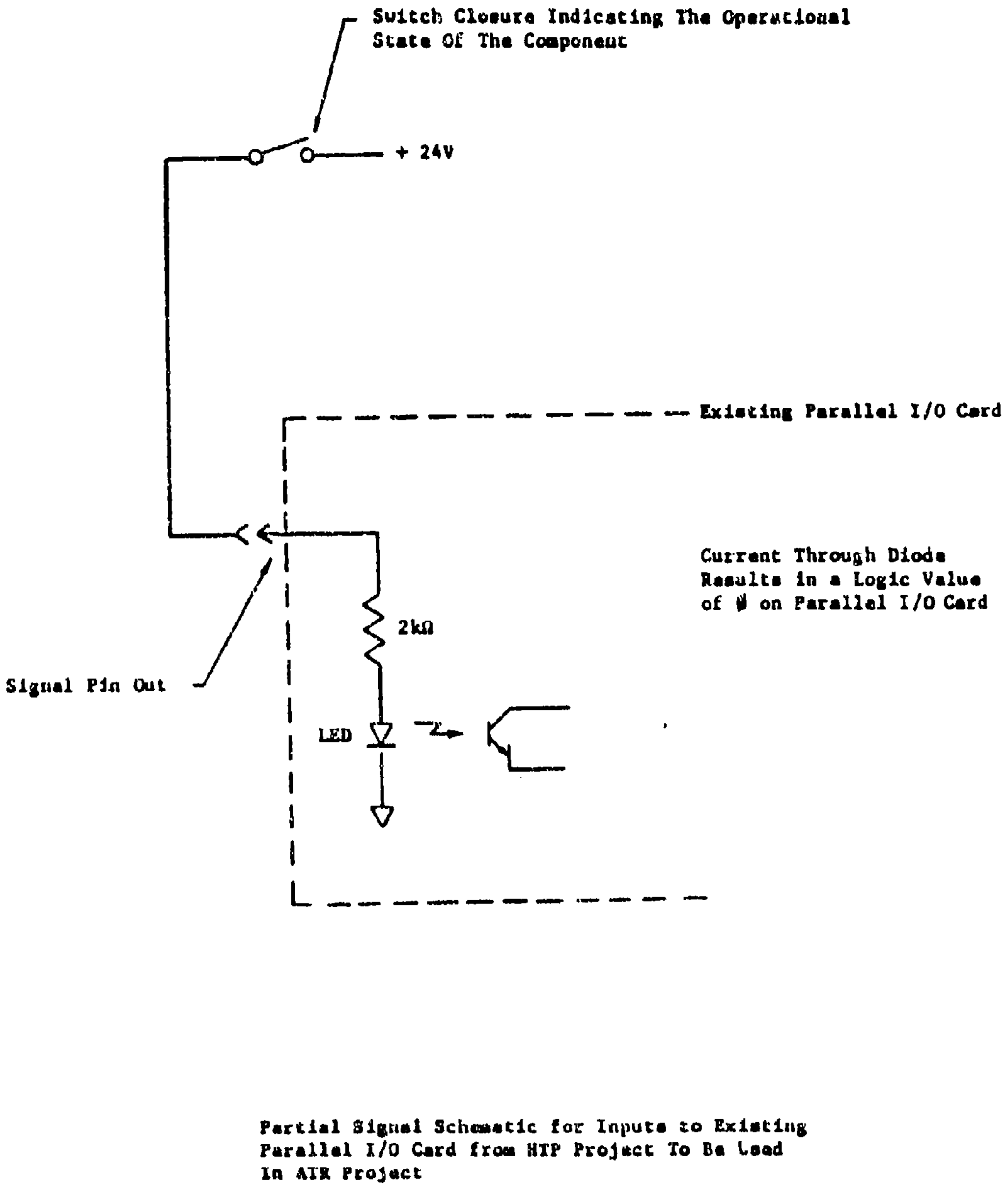

Switch Closure Indicating The Operational State Of The Component
+ 24V
Existing Parallel I/O Card
Current Through Diode Results in a Logic Value of Ø on Parallel I/O Card
2kΩ
Signal Pin Out
LED
Partial Signal Schematic for Inputs to Existing Parallel I/O Card from HTP Project To Be Used In ATR Project

Figure 2.1-13 Digital Input Signal Schematic as a Switch Closure to a Voltage Source

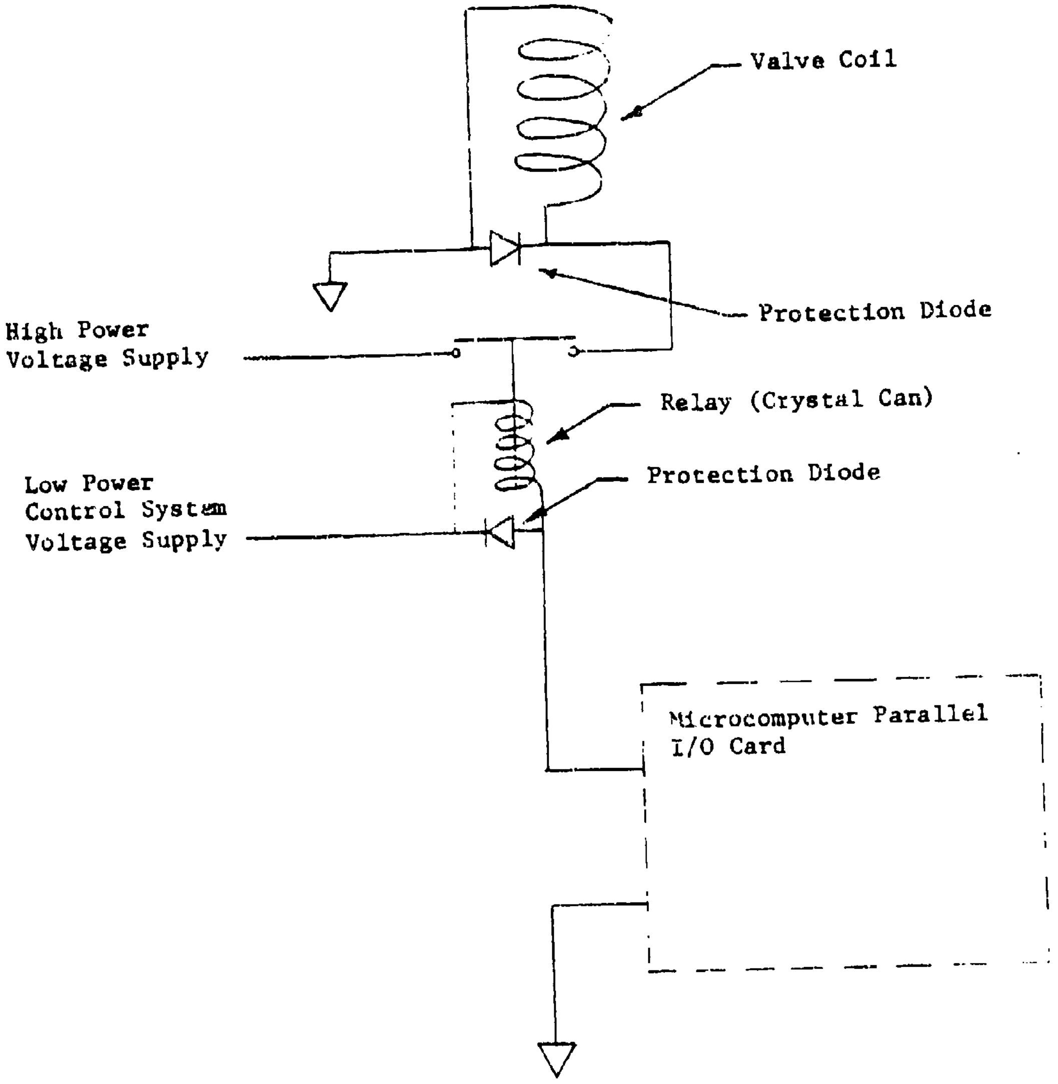


Figure 2.2.-1 Schematic For Typical Digital Output Signal

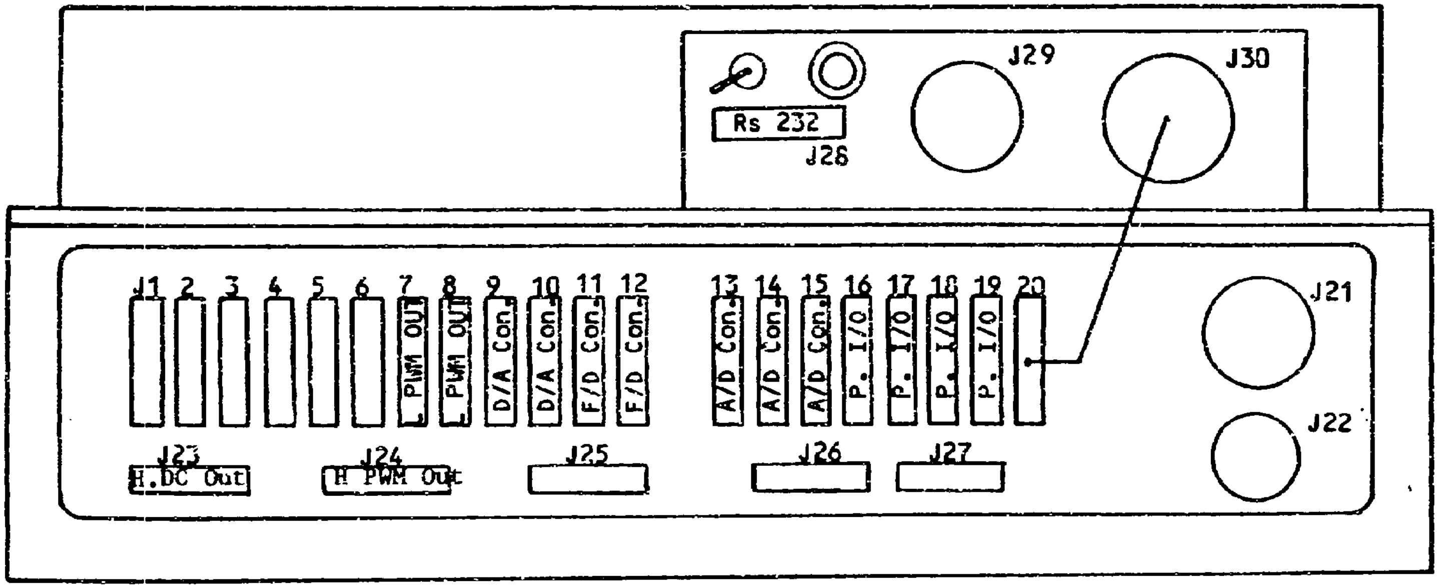


Figure 2.3-1 SC-1 Computer Connector Locations

The SC-1 uses the 5 MHz Intel 8086 central processing unit in unison with the Intel 8087 math processor and 8089 Input/Output processor, all very large-scale integration (VLSI) processors. All reside on the system's local bus. A block diagram of the computer is shown in Figure 2.3-2 and Table 2.3-1 summarizes the SC-1's general specifications.

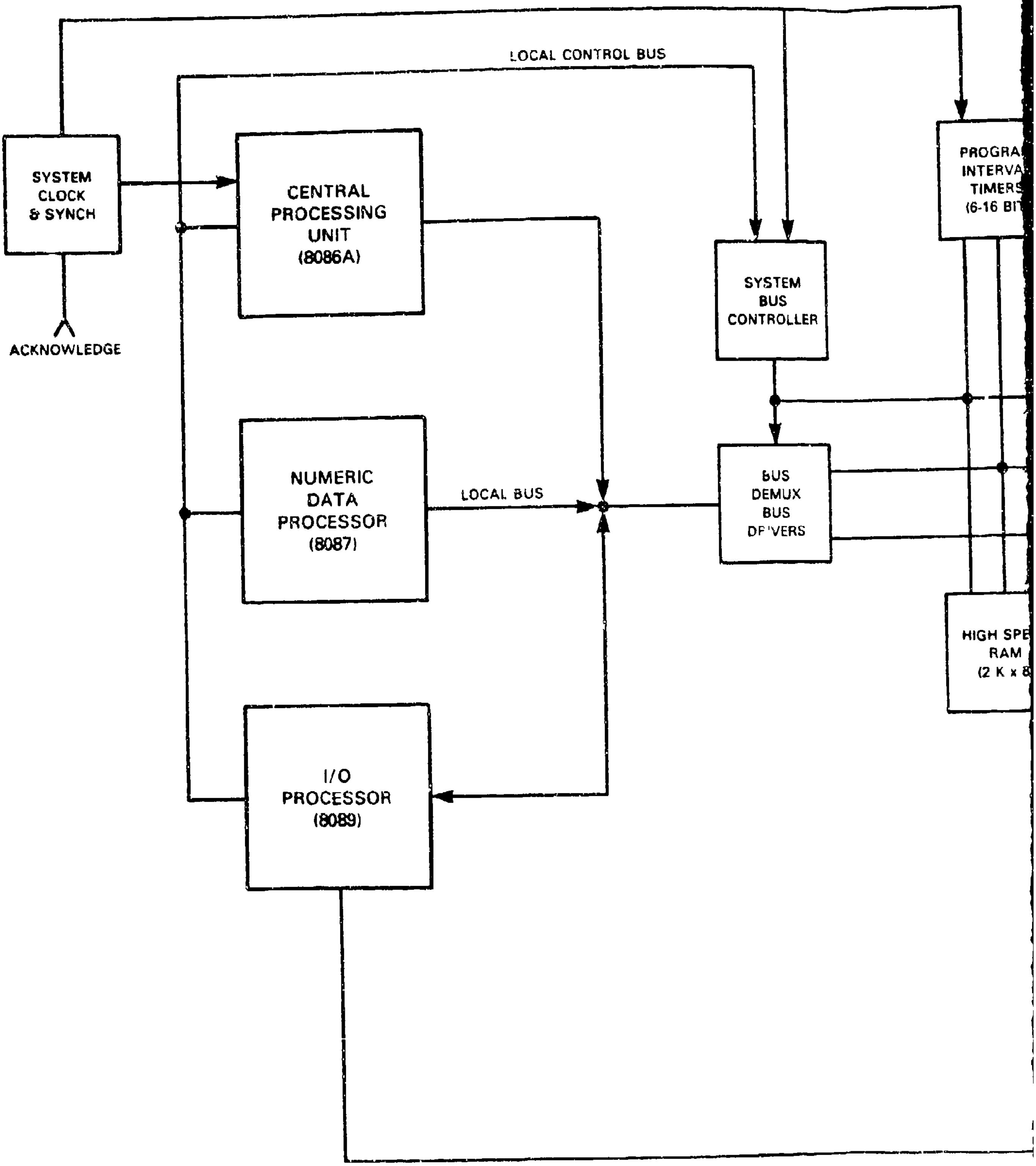
In its current configuration, the SC-1 uses a tri-processor configuration to provide increased throughput. Using the 8086's pipelined architecture, a great amount of parallel processing can be achieved. The 8087 is a purely numeric processor and can execute some numerical instructions 500 times faster than those instructions emulated from an 8086.

The Intel 8089 is a dedicated I/O processor. It communicates with an I/O expander unit which houses A/D, D/A and F/V converters, and a parallel I/O board. The 8089 reads vehicle inputs at the I/O expander and places them in main memory where the applications program can process them. It also reads a block of data from memory which represents control outputs and sends them to the I/O expander.

The SC/1 has three subsystems of memory. First, there is a 128K bubble memory unit to store the monitor and application programs. Second, there are also 2K bytes of static RAM for use as a system stack. Last, for main memory, there are 128K bytes of fault tolerant dynamic RAM. This DRAM provides single-bit failure detect/correct and multi-bit failure detect for the main memory.

To further mitigate the effects of a momentary software failure, a watchdog timer is provided in the circuit. The timer receives a signal every 100 msec from the software. If for some reason the software "lock up", a hardware reset is initiated and the program reboots from PROM. Execution restarts with a fresh copy of the software without the operator's intervention. A checksum over the application program provides a secondary verification. The checksum is computed every 100 msec. This checksum is compared to the original value sorted in PROM. If there is a discrepancy, the program has been inadvertently changed. Therefore, a hardware rest and software reload is initiated.

The SC-1's performance has been validated through many different environmental tests to make it acceptable to military and commercial applications. It has operated successfully through vibration tests to warrant use on



*Block diagram of SC-1 computer.*

Figure 2.3-

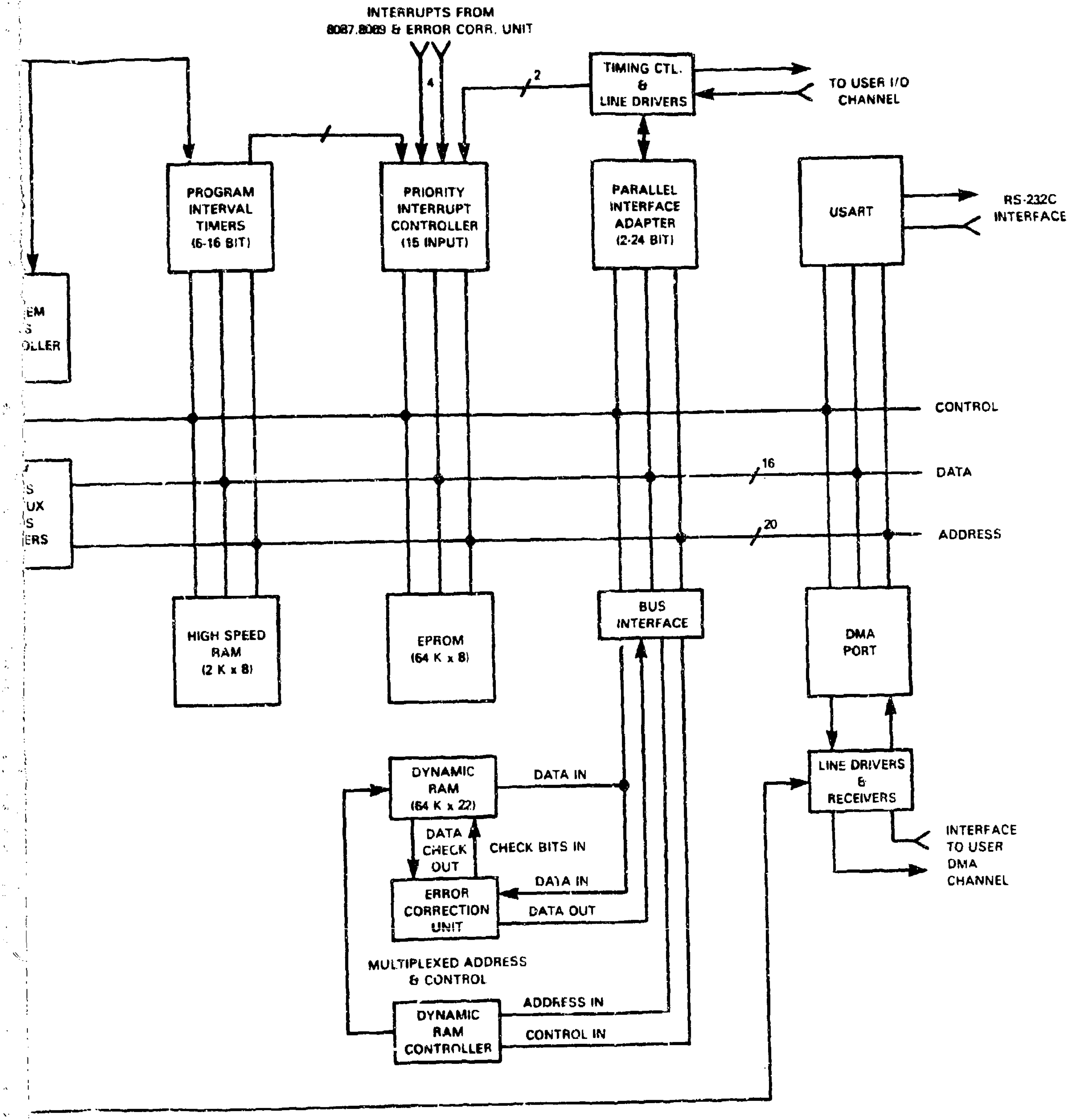


Figure 2.3-2 Block Diagram of the SC-1 Computer

## Table 2.3-1 General Specifications for SC-1 Computer

### Configuration

8086/8087/8089 triprocessor on local bus

### Word Size

Instructions: 8, 16, 24 or 32 bits
Data: 8, 16 bits (single word = 16 bits)

### Cycle Time

Basic instruction cycle: 0.8 s (instruction not in queue)

### Memory Capacity

Onboard Bubble: 128K bytes (expandable to 256K)
Onboard DRAM: 128K bytes (error correcting single-bit detect/correct; multi-bit detect)
Onboard SRAM: 2K bytes

### I/O Capacity

Parallel: 48 lines programmable (8255s), using two parallel interface adapters (equipped with LSI controller to emulate IBM-360 I/O channel handshaking).
DMA: Two 16-bit DMA ports, at 1M-bytes max transfer rate
Serial: RS-232 port, controlled by USART for both standard asynchronous or synchronous (8251A) communications

### Interrupts

Two 8-input priority interrupt controllers (15 hardware vectored interrupt lines available). Software configured for input priorities and mode (8259As).

### Timer

Two timers, each equipped with three 16-bit interval timers. (Timer outputs available as interrupt inputs.) Software configured for mode and rate (8253s).

### Power Consumption

20W

### Weight

9.38 lb

any application. The SC-1 can operate in a pure vacuum and in the temperature range of -40° to +80°C. All power is dissipated through the base plate; no fan is required. The micro also has proven electromagnetic compatibility.

## 2.4 CONTROL SYSTEM HARDWARE CONFIGURATION

The hardware of the overall control system is illustrated in Figure 2.4-1. The system consists of the SC-1 Computer, the Milltope Recorder, the Terra computer, junction boxes 1, 2, and 3, and the interconnecting wiring.

Southwest Research Institute Drawing No. A10048 shows the main power, ground, and shielding electrical schematic within Junction Box 1. Hydraulic, automotive, and control system power and ground a.e received through connectors J42-J44 of Junction Box 1. From these connections, the power and ground sources are distributed within Junction Box 1 via Deutsch connectors.

Appendix B of this report presents individual circuit schematics for all of the signals presented in Table 2.0-1. These schematics have been prepared as a design and diagnostics aid for the system checkout and field diagnostics. The power, ground, and shielding portions of these individual circuit diagrams are developed to the point where the connect with the distribution hardware presented in Figure 2.3-1. This must be noted when reviewing each individual circuit.

A completed set of electrical schematics and stick-line drawings are provided under separate cover. These drawings include the following:

- o Computer to computer enclosure interconnections
- o Computer enclosures to Junction Box 1, and Terra computer, and Miltope recorder interconnections
- o Junction Box 1 Schematic, Dwg. No. A10048
- o Junctions Box 1 to engine compartment sensors and actuators, and to Junction Boxes 2 and 3 interconnections
- o Junction Box 2 Schematic
- o Junction Box 2 to aft vehicle drivetrain sensor and actuator interconnections
- o Junction Box 3
- o Junction Box 3 to mid-vehicle drivetrain sensor and actuator interconnections.

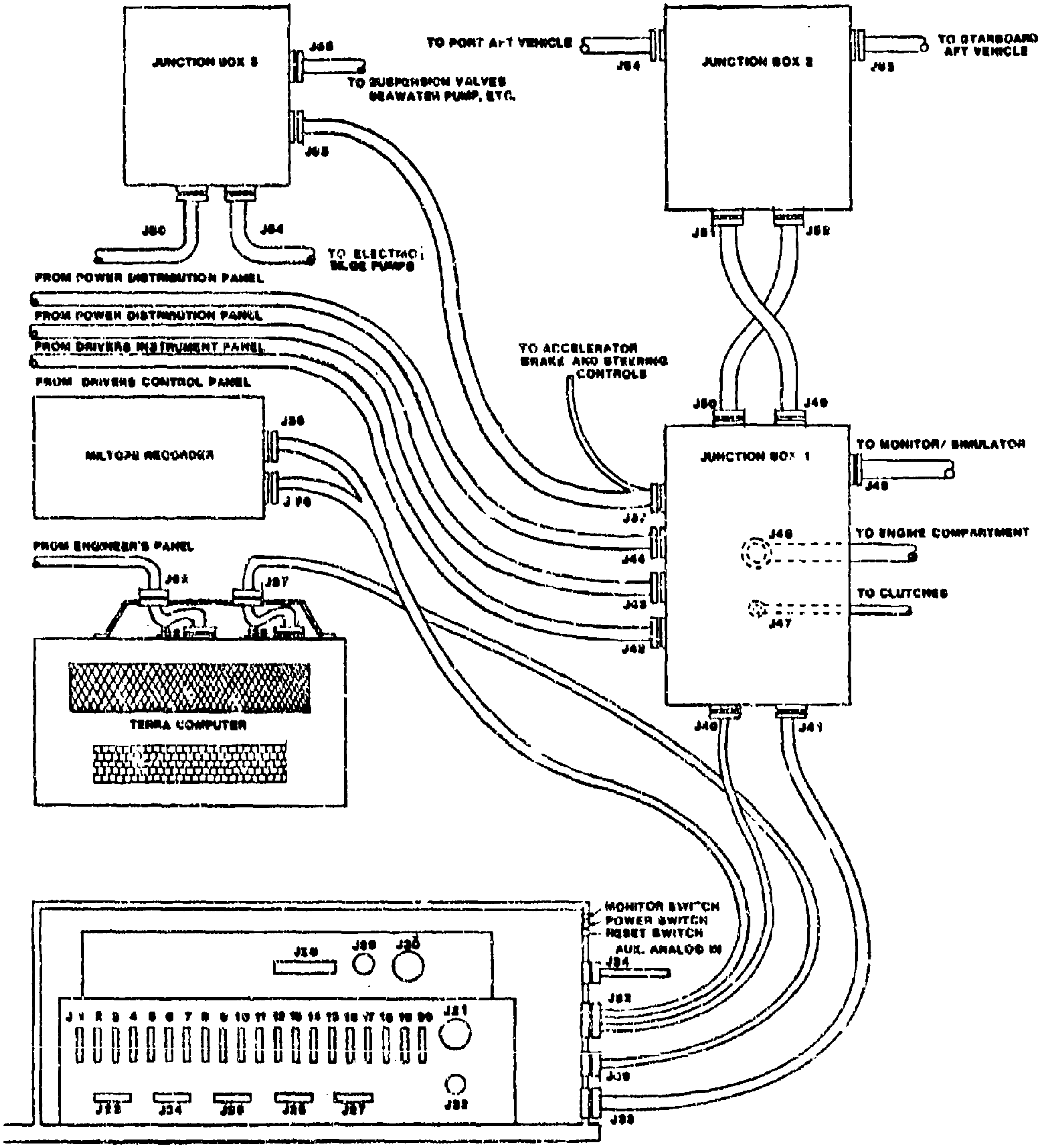


Figure 2.4-1 ATR Control System
Hardware Configuration

# 3.0 ATR SOFTWARE DESCRIPTION

## 3.1 ATR SOFTWARE OVERVIEW AND FLOWCHART

The software for the ATR is divided into five parts as follows:

- o Initialization
- o MAIN Program
- o Input/Conversion Subroutines
- o Calculation Subroutines
- o Output Subroutines

The overall structure of the ATR software is designed to avoid recursive subroutine calls, GOTO's, and bit tests (where the result is not obvious) through the use of logical variables (.TRUE. .FALSE.). In addition, logical variables are used in the MAIN program structure to reference when subroutines such as SHIFT and CHOMP have been called.

Figure 3.1-1 provides the main flowchart for the ATR program. Each of the program segments are idnividually described in the following sections.

### 3.1.1 Initialization

The function of the initialization section is to set all error flags to .FALSE., configure the vehicle to the transition mode and to suppress the display of error messages until the vehicle is operating.

Called from: Nowhere (part of ATR MAIN program)

Called when: Vehicle is first started or when it is determined that the vehicle's engine has stopped running.

Sequence of Events

When power is first turned on or the engine has stopped running:
  Set error flags to .FALSE.
  Set all other variables to initial values.
When the engine has started:
  Turn off hydraulic by-pass valve.
  Output standard wait message.
  Configure vehicle to transition mode.
  Wait 30 seconds for the suspension to extend.

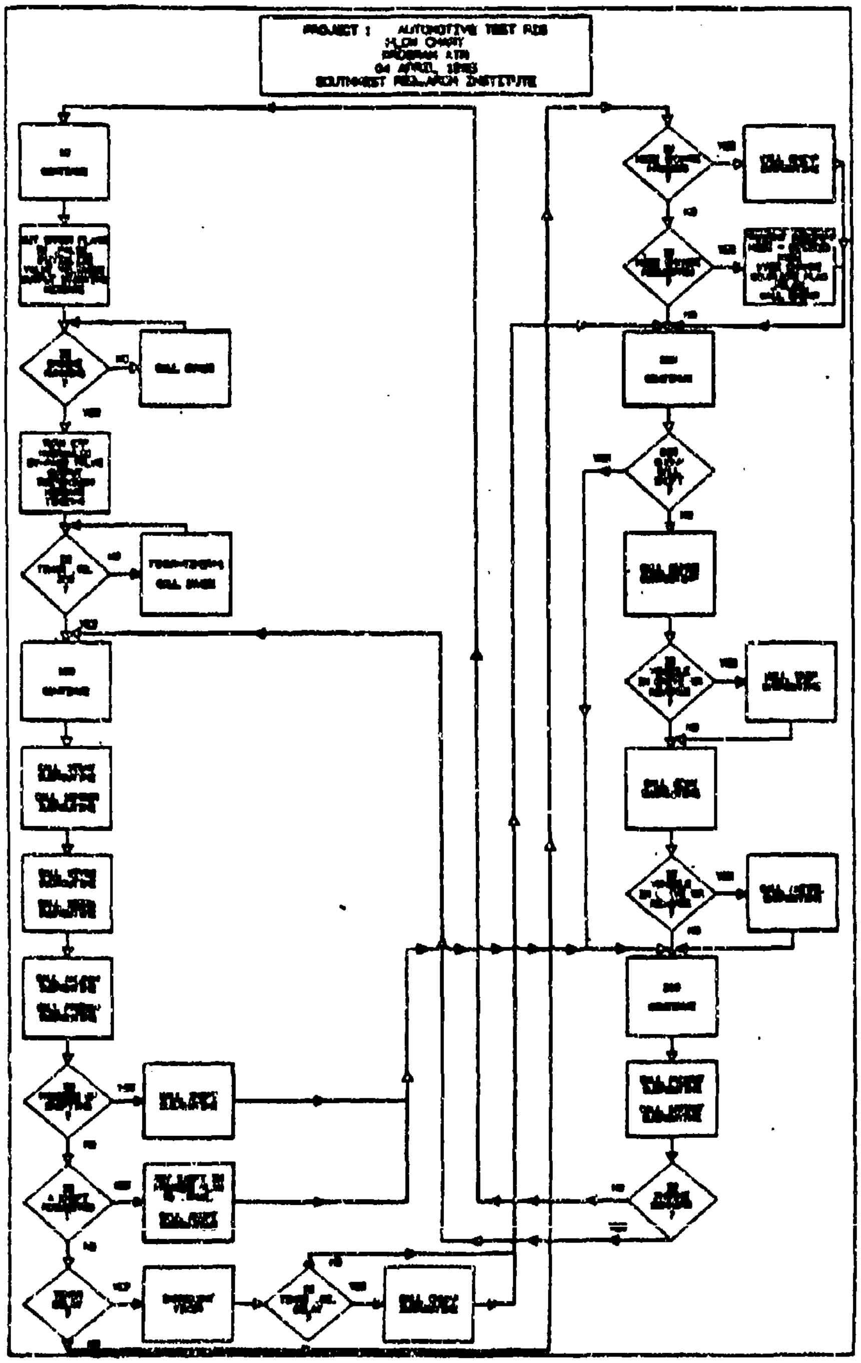


Figure 3.1-1 Main Flow Chart For ATR Program

3.1.2 MAIN

The ATR MAIN program functions as a driver for the subroutines contained within the ATR main loop. The ATR main loop is defined as those subroutines contained within the input/conversion, calculation, and output sections of the program.

Called from: Nowhere (starts execution when the vehicle is started)

Called when: The SC-1 is turned on

Sequence of Events

Call SEGMT.
Call SENGER.
Call SEVOS.
Call SEASA.
Call ANLGIN.
Call FREQIN.

Are we presently shifting? Yes, Call SHIFT.
Is a shift Requested? Yes, Call SHIFT.
Has CHOMP set a timer delay? Yes, if delay finished call CHOMP.
Are we presently changing modes? Yes, Call CHOMP.
Is a mode change requested? Yes, Call CHOMP.
Did CHOMP Call SHIFT? Yes, continue at the output section.
Call DEMOS.
Is the vehicle in drive or reverse? Yes, Call TREP.
Call ECAN.
Is the vehicle in drive or reverse? Yes, call ANTSTL.
Call PWMOUT.
Call ERROUT.
Is the engine still running? Yes, loop back to Call SEGMT.
No, loop back to initialization.

3.1.3 Input/Conversion Subroutines

These subroutines have the task of decoding and error checking the digital inputs in the SEGMT, SENGER, SEVOS, and SESEA subroutines, And converting to engineering units and error checking the analog inputs in the ANLGIN and FREQIN subroutines.

### 3.1.3.1 Subroutine SEGMT

The subroutine SEGMT (Set, Error check, Gear, Mode, Transmission) is used to set and error check the variables associated with the gear selector switches, the mode selector switches, and the transmission selector switches.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

Get from Common DIGIN(3)

Select from DIGIN(3) the bits to be checked for the Mode Switches and place the bits into TLTS.

Select from DIGIN(3) the bits to be checked for the Transmission Switches and place the bits into TPNRD

Select from DIGIN(3) the bits to be checked for the Gear Switches and place the bits into TDGS

Check to see if the Mode selector is in LAND, or TRNSTN, or SEA.

If Mode Selector is not in one of these positions then set the Mode Selector Switch Failure (MSSF) error flag to .TRUE.

Set a default value equal to Transition mode of operation.

Check to see of the transmission selector is in PARK or NTRAL or REVRSE or DRIVE.

If the Transmission Selector is not in one of these positions then set the Transmission Selector Switch Failure (TSSF) error flag to .TRUE.

Set a default value of DRIVE.

Check to see if Gear Selector is in either HIGH or LOW.

If Gear Selector is not in one of these positions then set the Gear Selector Switch Failure (GSSF) error flag to .TRUE.

### 3.1.3.2 Subroutine SENGER

The subroutine SENGER (Set ENGine ERrors) is used is used to check for failures in the vehicle's oil, hydraulic, and coolant systems.

Called from: MAIN

Called when: Each time through loop

Sequence of Events

Get from common (DIGIN(1).

Have SBITS select the bits associated with the following variables and place them in the variable TEST.

High Engine Oil Temperature Switch Status

Low Brake and Clutch Lube Pressure Switch Status

Low Brake Release Pressure Switch Status

Low Brake Supply Pressure Switch Status

Low Port Charge Pump Pressure Switch Status

Hydraulic Filter By-Pass Switch Status

Low Starboard Charge Pump Pressure Switch Status

Are any of the error bits .TRUE.?

If the answer is "no" then do not set the individual error flags.

If the answer is "yes" then set the individual error flags to their present values (.TRUE. or .FALSE.).

Set Low Brake and Clutch Lube Pressure Switch Failure (LBCLSF) error flag to present value.

Set Low Brake Release Pressure Switch Failure (LBRPSF) error flag to present value.

Set Low Brake Supply Pressure Switch Failure (LBSPSF) error flag to present value.

Set Hydraulic Filter By-PAss Switch Failure (HFBPSF) error flag to present value.

Set Low Starboard Charge Pump Pressure Switch Failure (LSCPSF) error flag to present value.

Ge from common DIGIN(2).

Low Engine Oil Pressure Switch Status

Low Port Hydraulic Oil Level Switch Status

Low Starboard Hydraulic Oil Level Switch Status

Are any of the error bits .TRUE.?

If the answer is "no" then do not set the individual error flags.

If the answer is "yes" then set the individual error flags to their present values (.TRUE. or .FALSE.).

Set Low Engine Oil Pressure Switch Failure (LEOPSF) error flag to present value.

Set Low Port Hydraulic Oil Level Switch Failure (LPHLSF) error flag to present value.

Set Low Starboard Hydraulic Oil Level Switch Failure (LSHLSF) error flag to present value.

Have SBITS select the bits associated with the following variables and place them in the variable TEST.

Low Scavenge Pump Pressure Status

Fire Sensed Switch Status

Set Low Scavenge Pump Pressure Failure (LSPPSF) error flag to present value.

Set Fire Sensed Switch Failure (FIRESF) error flag to present value.

If FIRESF .TRUE , close primary and secondary grills; turn off all fans.

3.1.3.3 Subroutine SEVOS

The SEVOS (Set Errors for VOltage and Suspension) subroutine checks for the failure of prot and starboard suspension components, along with checking for vehicle and control systems over or under voltages.

Called from: MAIN

Called when: Each time through loop

Sequence of Events

Get from common DIGIN(1).

Have SBITS select the bits associated with the following variables and place them in the variable TRPORT.

Port 1 Suspension Component Switch Status

Port 2 Suspension Component Switch Status

Port 3 Suspension Component Switch Status

Port 4 Suspension Component Switch Status

Port 5 Suspension Component Switch Status

Get from common DIGIN(2).

Have SBITS select the bits associated with the following variables and place them in the variable TSTRBD.
  Starboard 1 Suspension Component Switch Status
  Starboard 2 Suspension Component Switch Status
  Starboard 3 Suspension Component Switch Status
  Starboard 4 Suspension Component Switch Status
  Starboard 5 Suspension Component Switch Status
  Low Vehicle System Voltage Switch Status
  Low Control System Voltage Switch Status
  High Vehicle System Voltage Switch Status
  High Control System Voltage Switch Status

Are there any .TRUE. error bits in the variable TPORT?
If the answer is "no" then do not set the individual error flags to their present values (.TRUE. or .FALSE.).

Set Port 1 Suspension Component Switch Failure (P1SCSF) error flag to present value.
Set Port 2 Suspension Component Switch Failure (P2SCSF) error flag to present value.
Set Port 3 Suspension Component Switch Failure (P3SCSF) error flag to present value.
Set Port 4 Suspension Component Switch Failure (P4SCSF) error flag to present value.
Set Port 5 Suspension Component Switch Failure (P5SCSF) error flag to present value.
Are there any .TRUE. error bits in the variable TPORT?
If the answer is "no" then do not set the individual error flags.
If the answer is "yes" then set the individual error flags to their present values (.TRUE. or .FALSE.).
Set Starboard 1 Suspension Component Switch Failure (S1SCSF) error flag to present value.

Set Starboard 2 Suspension Component Switch Failure (S2SCSF) error flag to present value.

Set Starboard 3 Suspension Component Switch Failure (S3SCSF) error flag to present value.

Set Starboard 4 Suspension Component Switch Failure (S4SCSF) error flag to present value.

Set Starboard 5 Suspension Component Switch Failure (S5SCSF) error flag to present value.

Set Low Vehicle System Voltage Switch Failure (LVSVSF) error flag to present value.

Set Low Control System Voltage Switch Failure (HVSVSF) error flag to present value.

Set High Control System Voltage Switch Failure (HCSVSF) error flag to present value.

### 3.1.3.4 Subroutine SESEA

The subroutine SESEA (SEt SEA/land transition) determines the status of the bilge pumps, and ramp and grill closures. It also determines when the bilge pumps, ramp, and grill are in the proper position to change the mode of operation.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

Get DIGIN(3) from common.

Have SBITS select the bits associated with the following variables and place them in the variable CKPUMP.

AFT Electric Bilge Pump Activation Status

Forward Electric Bilge Pump Activation Status

Hydraulic Bilge Pump Valve Activation Status

Get DIGIN(2) from common.

Have SBITS select the bits associated with the following variables and place them in the variable CKRAMP.

Primary Grill Closed Switch Status
Ramp closed and Latched Switch Status

Set Aft Electric Bilge Pump Activation Status Flag (AEBPSF) to present value.

Set Seawater Pump Activation Status Flag (SEWPSF) to present value.

Set Hydraulic Bilge Pump Valve Activation Status Flag (HBPVSF) to present value.

Set Forward Electric Bilge Pump Activation Status Flag (FEBPSF) to present value.

Set Primary Grill Closed Switch Status Flag (PGCSSF) to present value.

Set Ramp Closed and latched Switch Status Flag (RCLSSF) to present value.

Are all pumps on and everything closed and locked?

If the answer is "yes" set the Land to Sea Transition Flag to .TRUE.

"No" continue.

Are the pumps off and the grill open?

If the answer is "yes" set t e Sea Land Transition Flag to .TRUE.

"No" continue.

### 3.1.3.5 Subroutine ANLGIN

The subroutine ANLGIN (ANaLoG INputs) converts the analog signals from the steering, brakes, accelerator, waterjet bucket positions, and pressure transducers. In addition, ANLGIN sets the error flags associated with these signals and generates default values as needed.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

Convert port and starboard bucket angles on error default: Ødegrees

Convert desired high motor speed on error high default: present value

Convert desired high motor speed on error low default: Ø rpm

Convert high motor speed ratio (brake pedal) on error default: Ø(brakes fully applied)

Convert desired turn ratio on error default: 1 (straight ahead)

Conver motor pressures on error high default: present value

Convert motor pressures on error low default: 3000 psi ($lb/in^2$)

Convert engine coolant temperature on error default: 240°F

Convert hydraulic oil temperatures on error default: 240°F

3.1.3.6 Subroutine FREQIN

The subroutine FREQIN (FREQuency INputs) converts the frequency signals from the engine, motor, sprocket, and waterjet magnetic pickups. In addition, FREQIN sets error flags associated with these signals and generates default values as needed.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

Convert engine, motor sprocket, and waterjet speeds.

Check for engine mag pickup failure.

Check for motor and sprocket and waterjet failures only when in drie or reverse.

Check for motor and sprocket failurs only when in land or transition modes.

Check for waterjet failures only when in transition or seaborne modes.

On error default value for motor failures is derived from sprocket speeds.

On error defalt value for sprocket failures is derived from motor speeds.

On error default value for waterjets in transition mode is from motor speeds.

On error default value for waterjets in seaborne mode is from either port or starboard waterjet.

### 3.1.4 Calculation Subroutines

These subroutines control the vehicle, such as handling mode changes (CHOMP), or for calculating values, such as total required engine power (TREP). The subroutines use the converted values from the input/conversion subroutines as well as any error or status flags deemed necessary.

#### 3.1.4.1 Subroutine SHIFT

The subroutine SHIFT determines when a change in final drie gear ratios is allowable and when allowable, performs the shifting sequence.

Called from: MAIN or CHOMP

Called when: Present Final Drive Gear Clutch does not equal Desired Final Drive Gear Clutch

#### 3.1.4.2 Subroutine CHOMP

The subroutine CHOMP (Change of Operation Mode) determines which of four mode changes is desired, (Land to Transition, Transition to Sea, etc.) and then calls the appropriate subroutine to accomplish that specific mode change.

Called from: MAIN

Called when:
- o Timer delay is complete
- o In the mode change process
- o A mode change is requested

Sequence of Events

If PResent Mode Of OPeration (PRMOOP) is equal to LAND and Present Desired Mode of Operation (PDMOOP) is equal to TRNSTN or SEA then Present Desired Mode of Operation must (PDMOOP) equal TRNSTN.

Is the desired mode change from LAND to TRNSTN?

If the answer is "yes," call subroutine LNDTRN (Land to Transition).

If the answer is "no," is the desired mode change from TRNSTN to LAND?

If the answer is "yes," call subroutine TRNLND (Transition to Land).

If the answer is "no," is the desired mode change from SEA to TRNSTN?

If the answer is "yes," call subroutine SEATRN (Sea to Transition).

If the answer is "no," is the desired mode change from TRNSTN to SEA?

If the answer is "yes" then call subroutine TRNSEA (Transition to Sea).

If the answer is "no," then Return.

It should be noted that if the subroutine LNDTRN is called, its point of RETURN in the CHOMP subroutine is at the check for the Transition to Land mode change. There is no risk of calling more than one subroutine each pass through CHOMP because the mode determination, error checking, and default setting has already been accomplished by the subroutine SEGMT. Also, there is no risk of calling a different mode change subroutine the second, third, etc., time through CHOMP because the Present Desired Mode of Operation (PDMOOP) is not updated until the mode change is complete. (Remember that CHOMP bases the decision on which mode change subroutine to call on the Present Desired Mode of Operation (PDMOOP).

### 3.1.4.3 Subroutine LNDTRN

The subroutine (LaND to TRansitioN) manages to Land to Transition mode change.

Called from: CHOMP

Called when: Land to transition mode change requested

Sequence of Events

Is the vehicle is high gear? Yes, call SHIFT.

Is the vehicle going too fast to start the change to TRNSTN mode?

If the answer is "yes," set the error flag to output "MOOCNC (Mode Of Operation Change No Complete) - excessive vehicle speed" message, RETURN.

If the answer is "no," check to see if the ramp is closed and latched.

If the answer is "no" then Return.

If the answer is "ye," then:

Shut off: Primary cooling fans
Secondary cooling fans
Waterjet by-pass valve

Turn on: Sea water pump
Aft electric bilge pump
Forward electric bilge pump
Hydraulic bilge pump
Waterjet no by-pass valve

Has the 5-second time delay necesary for the fans to slow down before grill closure, been executed?

If the answer is "no," set the Time Delay (TDELAY) value for the main loop to use, and RETURN.

If the answer it "yes," close both the primary and secondary grills.

Check the Land to Sea Transition (LSTRNS) flag (set by the subroutine SESEA) to see if the grills are closed and the seawater and bilge pumps are on, and the ramp is closed and latched.

If LSTRNS is .FALSE., the vehicle is not properly configured for the mode change. RETURN.

If LNTRNS is .TRUE. then proceed to finalize the mode change.

Turn on the Mode of Operation Change Complete (MOOPCC) light (.TRUE.).

Set Present Mode of Opration (PMOOP) equal to Present Desired Mode of Operation (PDMOOP).

Zero the Time DELAY (TDELAY).

RETURN.

### 3.1.4.4 Subroutine TRNLND

The subroutine TRNLND (TRansitioN to LaND) manages the Transition to Land mode change.

Called from: CHOMP

Called when: Transition to land mode change requested

Sequence of Events

Has the 3-second time delay necessary for the grills to open before starting the fans, been executed?

If the answer is "no," set the Time DELAY (TDELAY) value for the main loop to use, open the grills, and RETURN.

If the answer is "yes:"

| | |
|---|---|
| Turn on: | Primary cooling fan |
| | Waterjet by-pass valve |
| Turn off: | Seawater pump |
| | Electric bilge pumps |
| | Hydraulic bilge pump |
| | Waterjet no by-pass valve |

Check the Sea to Land Transition Flag (SLTRNS) (set by the subroutine SESEA) to see if the grills are open and the seawater and bilge pumps are off.

If SLTRNS is .FALSE., the vehicle is not properly configured for the mode change. RETURN.

If SLTRNS is .TRUE. then proceed to finalize the mode change.

Turn on the Mode of Operation Change Complete (MOOPCC) light. (.TRUE.)

Set Present Mode of Operation (PRMOOP) equal to Present Desired Mode of Operation (PDMOOP).

Zero the Time DELAY (TDELAY).

RETURN.

### 3.1.4.5 Subroutine TRNSEA

The subroutine TRNSEA (TRansitioN to SEA) manages the Transition to Sea mode change.

Called from: CHOMP

Called when: Transition to sea mode requested

Sequence of Events

Disengage high clutch.

Disengage low clutch.

Turn on: Motor by-pass valve.

Turn off: Motor no by-pass valve.

Has the 5-second delay necessary for the tracks to stop moving, been executed?

If the answer is "no," set the Time DELAY value for the main loop to use, and RETURN.

If the answer is "yes," raise the suspension and finalize the mode change.

Turn on Mode Of Operation Change Complete light (MOOPCC)(.TRUE.).

Set PResent Mode Of Operation (PRMOOP) equal to Present Desird Mode Of Operation (PDMOOP).

Zero the Time DELAY (TDELAY).

RETURN

### 3.1.4.6 Subroutine SEATRN

The subroutine SEATRN (SEA to TRansitioN) manages the Sea to Transition mode change.

Called from: CHOMP

Called when: Sea to transition mode change requested

Sequence of Events

Lower the suspension.

Has the 30-second delay necessary when lowering the suspension, been executed?

If the answer is "no," set the Time DELAY (TDELAY) value for the MAIN loop to use, and RETURN.

If the answer is "yes:"

- Engage low clutch
- Disengage high clutch.
- Turn on: Motor no by-pass valve
- Turn off: Motor by-pass valve

Turn on Mode of Operation Change Complete light (MOOPCC) (.TRUE.).

Set PResent Mode Of OPeration (PRMOOP) equal to Present Desired Mode Of Operation (PDMOOP).

Zero the Time DELAY (TDELAY).

RETURN

### 3.1.4.7 Subroutine DEMOS

The subroutine DEMOS (DEsired MOtor Speed) calculates desired port and starboard motor speeds from the accelerator pedal, brake pedal, and joystick signals.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

In landborne mode calculate port and starboard desired motor speeds based on turn direction.

In transition mode calculate port and starboard waterjet speeds based on turn direction.

In seaborne mode desired waterjet speed is the same as desired high waterjet speed (steering accomplished by buckets).

### 3.1.4.8 Subroutine TREP

The subroutine TREP (Total Required Engine Power) calculates the amount of power required by the transmissions. This is accomplished by having TREP determine if the vehicle is in the landborne mode or one of the other two modes calling either LDTREP or TSTREP.

Called from: MAIN

Called when: Vehicle is in drive or reverse

Sequence of Events

Is the vehicle in the landborne mode? Yes, Call LDTREP.

Is the vehicl ont in the landborne mode? Yes, Call TSTREP.

### 3.1.4.9 Subroutine LDTREP

The subroutine LDTREP (LanD TREP) calculates total required engine power when the vehicle is in landborne mode.

Called from: TREP

Called when: Vehicle is in landborne mode

Sequence of Events

Determine if the port pump or if the port motor is at maximum displacement.

Calculate port torque.

Calculate port horsepower.

Repeat the sequence for the starboard pump and motor.

### 3.1.4.10 Subroutine TSTREP

The subroutine TSTREP (Transition Sea TREP) calculates total required engine power when the vehicle is in either transition or seaborne modes.

Called from: TREP

Called when: Vehicle is in transition or seaborne modes.

Sequence of Events

Calculate port flow rate.

Calculate port waterjet power.

Calculate required power for the port transmission.

Repeat the sequence for the starboard transmission.

### 3.1.4.11 Subroutine ECAN

The subroutine ECAN (Engine Control and ANalysis) determines governor settings and controls the cooling system fans.

Called from: MAIN

Called when: Each time through the loop

### 3.1.4.12 Subroutine ANTSTL

The subroutine ANTSTL (ANTi STalL) insures that the hydrostatic drivetrain does not try to extract more engine power than the engine can produce at that moment.

Called from: MAIN

Called when: Vehicle is in drive or reverse

Sequence of Events

Calculate port and starboard pump power.

If in landborne mode: Calculate maximum motor speed;
Compare maximum with desired;
Turn direction and turn ratio then determines allowable port and starboard motor speeds

If in transition mode: Calculate maximum waterjet speed:
Compare maximum with desired;
Turn direction and turn ratio then determines allowable port and starboard waterjet speeds

If in seaborne mode: Calculate maximum waterjet speed;
Compare maximum with desired;
Allowable port and starboard waterjet are the same (bucket steering)

### 3.1.5 Output Subroutines

These subroutines apply the control signals to the individual control elements as well as communicate the error messages to the Terra computer. The output signals consist of Pulse Width modulated signals and discreet on/off signals. The communication of the error messages is handled via transmission of four 16-bit words to the Terra via RS 232 communication. The error words are updated and stored in common.

### 3.1.5.1 Subroutine ERROUT

The subroutine ERROUT (ERRor OUTput) combines the individual error flags into four sixteen-bit digital words. In addition, ERROUT determines if that particular error is proper for the present mode of operation.

Called from: MAIN

Called when: Each time through the loop

Sequence of Events

Is the error flag .TRUE.? Yes, put a one in the appropriate bit position.

Suppress errors generated in SESEA subroutine until deemed appropriate.

If between land and transition modes and the vehicle is going too fast, generate that error.

### 3.1.5.2 Subroutine PWMOUT

This subroutine converts the desired transmission, engine, and clutch valve voltages into appropriate integer values which are then placed in Common to be transmitted to the PWM output cards.

Called from: MAIN

Called when: Each time through loop

Sequence of Events

Desired voltage values are converted to integer value duty cycle equivalent values and placed in appropriate Common block locations for each control device.

## 3.2 ATR Variable and Error Message Listing

The ATR variable listing and error message listing is provided in Table 3.2-1.

### 3.2.1 Variable Listing

The variable listing provides the formal variable name that is used in the software and a descriptive title associated with each variable name. This is presented as Table 3.2-1.

### 3.2.2 Error Message Listing

Table 3.2-2 presents the listing of the error messages which have been used for the ATR vehicle. This listing relates the word number and bit number associated with each error message. Also presented in this table is an error number assigned to each error and the actual error message that is presented on the Terra computer. Lastly, this table presents the name of the subroutine which detects the error and the program variable name that is used to convey the status of the error.

Table 3.2-1 ATR Program Variable Listing

| | |
|---|---|
| AEBPSF | Error Flag |
| AENSF | Error Flag |
| AENSP | Actual Engine Speed |
| ALPMSP | Allowable Port Motor Speed |
| ALPWSP | Allowable Port Waterjet Speed |
| ALSMSP | Allowable Starboard Motor Speed |
| ALSWSP | Allowable Starboard Wateret Speed |
| APBCAN | Actual Port Bucket Angle |
| APBCSF | Error Flag |
| EPMSF | Error Flag |
| APMSP | Actual Port Motor Speed |
| APPS | Actual Port Pump Speed |
| APSSF | Error Flag |
| APWSF | Error Flag |
| ASBCAN | Actual Starboard Bucket Angle |
| ASBCSF | Error Flag |
| ASMSF | Error Flag |
| ASMSP | Actual Starboard Motor Speed |
| ASPS | Actual Starboard Pump Speed |
| ASSSF | Error Flag |
| ASWSF | Error Flag |
| BIT | Bit pointer for TBIT function call |
| BIT1 | Bit pointer for TBIT function call |
| BIT10 | Bit pointer for TBIT function call |
| BIT11 | Bit pointer for TBIT function call |
| BIT12 | Bit pointer for TBIT function call |
| BIT13 | Bit pointer for TBIT function call |
| BIT14 | Bit pointer for TBIT function call |
| BIT15 | Bit pointer for TBIT function call |
| BIT16 | Bit pointer for TBIT function call |
| BIT17 | Bit pointer for TBIT function call |
| BIT18 | Bit pointer for TBIT function call |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| BIT19 | Bit pointer for TBIT function call |
| BIT2 | Bit pointer for TBIT function call |
| BIT20 | Bit pointer for TBIT function call |
| BIT21 | Bit pointer for TBIT function call |
| BIT22 | Bit pointer for TBIT function call |
| BIT23 | Bit pointer for TBIT function call |
| BIT24 | Bit pointer for TBIT function call |
| BIT25 | Bit pointer for TBIT function call |
| BIT26 | Bit pointer for TBIT function call |
| BIT27 | Bit pointer for TBIT function call |
| BIT28 | Bit pointer for TBIT function call |
| BIT29 | Bit pointer for TBIT function call |
| BIT3 | Bit pointer for TBIT function call |
| BIT30 | Bit pointer for TBIT function call |
| BIT31 | Bit pointer for TBIT function call |
| BIT32 | Bit pointer for TBIT function call |
| BIT33 | Bit pointer for TBIT function call |
| BIT34 | Bit pointer for TBIT function call |
| BIT35 | Bit pointer for TBIT function call |
| BIT36 | Bit pointer for TBIT function call |
| BIT37 | Bit pointer for TBIT function call |
| BIT38 | Bit pointer for TBIT function call |
| BIT39 | Bit pointer for TBIT function call |
| BIT4 | Bit pointer for TBIT function call |
| BIT40 | Bit pointer for TBIT function call |
| BIT41 | Bit pointer for TBIT function call |
| BIT42 | Bit pointer for TBIT function call |
| BIT43 | Bit pointer for TBIT function call |
| BIT44 | Bit pointer for TBIT function call |
| BIT45 | Bit pointer for TBIT function call |
| BIT46 | Bit pointer for TBIT function call |
| BIT47 | Bit pointer for TBIT function call |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| BIT48 | Bit pointer for TBIT function call |
| BIT49 | Bit pointer for TBIT function call |
| BIT5 | Bit pointer for TBIT function call |
| BIT6 | Bit pointer for TBIT function call |
| BIT7 | Bit pointer for TBIT function call |
| BIT8 | Bit pointer for TBIT function call |
| BIT9 | Bit pointer for TBIT function call |
| BLGOFF | Bit pattern for Bilge Pumps Off |
| BLGPON | Bit pattern for Bilge Pumps On |
| CKPLST | Check Pump Land/Sea Transition |
| CKPSLT | Check Pump Sea/Land Transition |
| CKPUMP | Check Pump |
| CKRAMP | Check Ramp |
| CKRLST | Check Ramp Land/Sea Transition |
| CKRSL1 | Check Ramp Sea/Land Transition 1 |
| CKRSL2 | Check Ramp Sea/Land Transition 2 |
| DES | J7 P3 Desired Engine Speed |
| DHCVV | J7 P1 Desired High Clutch Valve Voltage |
| DHMHSF | Error Flag |
| DHMLSF | Error Flag |
| DHMRSF | Error Flag |
| DHMSP | Desired High Motor Speed |
| SHMSR | Desired High Motor Speed Ratio |
| DHWSP | Desired High Waterjet Speed |
| DHWSR | Desired High Waterjet Speed Ratio |
| DIGIN | Digital Input (3-element array) |
| DIGOUT | J7 P2 Digital Output (7-element array) |
| DLCVV | Desired Low Clutch Valve Voltage |
| DMOOP | Desired Mode of Operation |
| DPFRT | Desired Port Flow Rate |
| DPMP | Desired Port Motor Power |
| DPMSP | Desired Port Motor Speed |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| DPPP | Desired Port Pump Power |
| DPPWJ | Desired Power Port Waterjet |
| DPSWJ | Desired Power Starboard Waterjet |
| DPWSP | Desired Port Waterjet Speed |
| DRIVE | Drive |
| DSFRT | Desired Starboard Flow Rate |
| DSMP | Desired Starboard Motor Power |
| DSMSP | Desired Starboard Motor Speed |
| DSPP | Desired Starboard Pump Power |
| DSWSP | Desired Starboard Waterjet Speed |
| DTRNR | Desired Turn Ratio |
| DTRNSF | Error Flag |
| DTRST | Desired Transmission Setting |
| ENCTEM | Engine Coolant Temperature |
| ENCTSF | Error Flag |
| END | |
| ERRDLY | Error Delay |
| FDR | Final Drive Ratio |
| FEBPSF | Error Flag |
| FIRESF | Error Flag |
| GCTD | Grill Close Time Delay |
| GOTD | Grill Open Time Delay |
| GRCLSE | Bit pattern for Grill Closure |
| GROPEN | Bit pattern for Grill Open |
| GSSF | Error Flag |
| HBPVSF | Error Flag |
| HCSVSF | Error Flag |
| HEOTSF | Error Flag |
| HIGH | High Gear |
| HVSVSF | Error Flag |
| IAENSP | J12 P18 Input Actual Engine Speed |
| IAPBCN | J13 P4 Input Actual Port Bucket Angle |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| IAPMSP | J11 P16 Input Actual Port Motor Speed |
| IAPSSP | J11 P18 Input Actual Port Sprocket Speed |
| IAPWSP | J12 P16 Input Actual Port Waterjet Speed |
| IASBCN | J13 P5 Input Actual Starboard Bucket Angle |
| IASMSP | J11 P17 Input Actual Starboard Motor Speed |
| IASSSP | J11 P19 Input Actual Starboard Sprocket Speed |
| IASWSP | J12 P17 Input Actual Starboard Waterjet Speed |
| IDHMSP | J13 P1 Input Desired High Motor Speed |
| IDHMSR | J13 P2 Input Desired High Motor Speed Ratio |
| IDTRNR | J13 P3 Input Desired Turn Radius |
| IENCTM | J14 P7 Input Engine Coolant Temperature |
| INTCPT | Intercept |
| IPAMPR | J14 P3 Input Port Aft Motor Pressure |
| IPFMPR | J14 P1 Input Port Forward Motor Pressure |
| IPHOTM | J14 P5 Input Port Hydraulic Oil Temperature |
| ISAMPR | J14 P4 Input Starboard Aft Motor Pressure |
| ISFMPR | J14 P2 Input Starboard Forward Motor Pressure |
| ISHOTM | J14 P6 Input Starboard Hydraulic Oil Temperature |
| JCAR10 | Parallel I/O card pointers for SBIT function calls |
| JCAR11 | Parallel I/O card pointers for SBIT function calls |
| JCAR12 | Parallel I/O card pointers for SBIT function calls |
| JCAR13 | Parallel I/O card pointers for SBIT function calls |
| JCAR14 | Parallel I/O card pointers for SBIT function calls |
| JCAR15 | Parallel I/O card pointers for SBIT function calls |
| JCAR16 | Parallel I/O card pointers for SBIT function calls |
| JCAR9 | Parallel I/O card pointers for SBIT function calls |
| JCARD1 | Parallel I/O card pointers for SBIT function calls |
| JCARD2 | Parallel I/O card pointers for SBIT function calls |
| JCARD3 | Parallel I/O card pointers for SBIT function calls |
| JCARD4 | Parallel I/O card pointers for SBIT function calls |
| JCARD5 | Parallel I/O card pointers for SBIT function calls |
| JCARD6 | Parallel I/O card pointers for SBIT function calls |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| JCARD7 | Parallel I/O card pointers for SBIT function calls |
| JCARD8 | Parallel I/O card pointers for SBIT function calls |
| LAND | Landborne |
| LBCLSF | Error Flag |
| LBRPSF | Error Flag |
| LBSPSF | Error Flag |
| LCSVSF | Error Flag |
| LEOPSF | Error Flag |
| LOW | Low Gear |
| LPCPSF | Error Flag |
| LPHLSF | Error Flag |
| LSCPSF | Error Flag |
| LSHLSF | Error Flag |
| LSPPSF | Error Flag |
| LSTRNS | Land/Sea Transition Flag |
| LVSVSF | Error Flag |
| MASK | 16-element array that is used to build masks for TBIT, SBIT, and ERROUT |
| MAXMSP | Maximum Motor Speed |
| MAXWSP | Maximum Waterjet Speed |
| MBPON | Motor Bypass On |
| MLTSSF | Error Flag |
| MNBPON | Motor No Bypass On |
| MOOPCC | Mode of Operation Change Complete |
| MSSF | Error Flag |
| NAME | Name of variable that SBITS operates on |
| NTRAL | Neutrla |
| P1SCSF | Error Flag |
| P2SCSF | Error Flag |
| P3SCSF | Error Flag |
| P4SCSF | Error Flag |
| P5SCSF | Error Flag |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| PAMHSF | Error Flag |
| PAMLSF | Error Flag |
| PAMTPR | Port Aft Motor Pressure |
| PARK | Park |
| PATVV | J8 P3 Port Aft Transmission Valve Voltage |
| PCFWBP | Primary Cooling Fan, Waterjet Bypass |
| PCFWNP | Primary Cooling Fan, Waterjet No Bypass |
| PDIFP | Port Differential Pressure |
| PDMOOP | Present Desired Mode of Operation |
| PFMHSF | Error Flag |
| PFMLSF | Error Flag |
| PFMTPR | Port Forward Motor Pressure |
| PFTVV | J8 P1 Port Forward Transmission Valve Voltage |
| PGCSSF | Error Flag |
| PHOTEM | Port Hydraulic Oil Temperature |
| PHOTSF | Error Flag |
| PMDFP | Port Motor Differential Pressure |
| PMDIS | Port Motor Displacement |
| PMTRQ | Port Motor Torque |
| PPDFP | Port Pump Differential Pressure |
| PPDIS | Port Pump Displacement |
| PPMEFF | Port Pump Mechanical Efficiency |
| PPTRQ | Port Pump Torque |
| PRMOOP | Present Mode of Operation |
| PTREFF | Port Transmission Efficiency |
| RCLSSF | Error Flag |
| REPP | |
| REPPP | |
| REPPT | |
| REPSP | |
| REPST | |
| REPT | |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| RES | J9 P1 Real Engine Speed |
| RESULT | Test variable for TBIT function |
| REVRSE | Reverse |
| RMASK | Mask word for SBITS function |
| RPMS | J9 P4 |
| RPSS | J9 P10 |
| RPMS | J9 P5 |
| RSSS | J9 P11 |
| S1SCSF | Error Flag |
| S2SCSF | Error Flag |
| S3SCSF | Error Flag |
| S4SCSF | Error Flag |
| S5SCSF | Error Flag |
| SAMHSF | Error Flag |
| SAMLSF | Error Flag |
| SAMTPR | |
| SATVV | J8 P4 Starboard Aft Transmission Valve Voltage |
| SDIFP | Starboard Differential Pressure |
| SEA | Seaborne |
| SECFOF | Secondary Fan Off |
| SEWPSF | Error Flag |
| SFMHSF | Error Flag |
| SFMLSF | Error Flag |
| SFMTPR | |
| SFTVV | J8 P2 Starboard Forward Transmission Valve Voltage |
| SHOTEM | Starboard Hydraulic Oil Temperature |
| SHOTSF | Error Flag |
| SLOWER | Suspension Lower bit pattern |
| SLTRNS | Sea/Land Transition Flag |
| SMDFP | Starboard Motor Differential Pressure |
| SMDIS | Starboard Motor Displacement |
| SMTRQ | Starboard Motor Torque |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| SPDFP | Starboard Pump Differential Pressure |
| SPDIS | Starboard Pump Displacement |
| SPMEFF | Starboard Pump Mechanical Efficiency |
| SPTRQ | Starboard Pump Torque |
| STAR10 | Starting bit pointer for SBIT function |
| STAR11 | Starting bit pointer for SBIT function |
| STAR12 | Starting bit pointer for SBIT function |
| STAR13 | Starting bit pointer for SBIT function |
| STAR14 | Starting bit pointer for SBIT function |
| STAR15 | Starting bit pointer for SBIT function |
| STAR16 | Starting bit pointer for SBIT function |
| STAR9 | Starting bit pointer for SBIT function |
| START | Starting bit pointer for SBIT function |
| START1 | Starting bit pointer for SBIT function |
| START2 | Starting bit pointer for SBIT function |
| START3 | Starting bit pointer for SBIT function |
| START4 | Starting bit pointer for SBIT function |
| START5 | Starting bit pointer for SBIT function |
| START6 | Starting bit pointer for SBIT function |
| START7 | Starting bit pointer for SBIT function |
| START8 | Starting bit pointer for SBIT function |
| STREFF | Starboard Transmission Efficiency |
| SUDOWN | |
| SURISE | Bit pattern to raise suspension |
| TBIT | Function name |
| TDELAY | Time Delay |
| TEST1 | Temporary variable for SENGER |
| TEST2 | Temporary variable for SENGER |
| TEST3 | Temporary variable for SENGER |
| TIMER | Main loop time delay counter |
| TRKSTP | Time delay for Truck Stop |
| TRNDIR | Turn Direction |

Table 3.2-1 ATR Program Variable Listing (Continued)

| | |
|---|---|
| TRNSTN | Transition |
| TSSF | Error Flag |
| WIDT10 | Field width pointers for SBIT function |
| WIDT11 | Field width pointers for SBIT function |
| WIDT12 | Field width pointers for SBIT function |
| WIDT13 | Field width pointers for SBIT function |
| WIDT14 | Field width pointers for SBIT function |
| WIDT15 | Field width pointers for SBIT function |
| WIDT16 | Field width pointers for SBIT function |
| WIDT9 | Field width pointers for SBIT function |
| WIDTH | Field width pointers for SBIT function |
| WIDTH1 | Field width pointers for SBIT function |
| WIDTH2 | Field width pointers for SBIT function |
| WIDTH3 | Field width pointers for SBIT function |
| WIDTH4 | Field width pointers for SBIT function |
| WIDTH5 | Field width pointers for SBIT function |
| WIDTH6 | Field width pointers for SBIT function |
| WIDTH7 | Field width pointers for SBIT function |
| WIDTH8 | Field width pointers for SBIT function |
| WJCNST | Waterjet Constant |

Table 3.2-2 Error Message Listing

| Word # | Bit # | Error # | Error Message | Subroutine | Variable Name |
|---|---|---|---|---|---|
| 3 | 4 | 36 | ACCEL PEDAL SENSOR HIGH FAILURE | ANLGIN | (DHMHSF) |
| 3 | 5 | 37 | ACCEL PEDAL SENSOR LOW FAILURE | ANLGIN | (DHMLSF) |
| 2 | 14 | 30 | AFT ELECTRIC BILGE PUMP FAILURE | SESEA | (AEBPSF) |
| 1 | 4 | 4 | BRAKES NOT APPLIED | SENGER | (BKNPSF) |
| 3 | 6 | 38 | BRAKE PEDAL POTENTIOMETER FAILURE | ANLGIN | (DHMRSF) |
| 4 | 6 | 54 | ENGINE MAG PICKUP FAILURE | FREQIN | (AENSF) |
| 1 | 5 | 5 | FIRE SENSED | SENGER | (FIRESF) |
| 2 | 15 | 31 | FORWARD ELECTRIC BILGE PUMP FAILURE | SESEA | (FEBPSF) |
| 1 | 1 | 1 | GEAR SELECTOR SWITCH FAILURE | SEGMT | (GSSF) |
| 1 | 15 | 15 | HIGH CONTROL SYSTEM VOLTAGE | SEVOS | (HCSVSF) |
| 3 | 7 | 39 | HIGH ENGINE COOLANT TEMPERATURE | ANLGIN | (ENCTSF) |
| 3 | 8 | 40 | HIGH PORT HYD OIL TEMPERATURE | ANLGIN | (PHOTSF) |
| 3 | 9 | 41 | HIGH STARBOARD HYD OIL TEMPERATURE | ANLGIN | (SHOTSF) |
| 1 | 16 | 16 | HIGH VEHICLE SYSTEM VOLTAGE | SEVOS | (HVSVSF) |
| 2 | 16 | 32 | HYDRAULIC BILGE PUMP FAILURE | SESEA | (HBPVSF) |
| 1 | 6 | 6 | HYDRAULIC FILTER BY-PASS FAILURE | SENGER | (HFBPSF) |
| 3 | 10 | 42 | INADVERTENT BRAKE APPLICATION | ANLGIN | (INBPSF) |
| 1 | 7 | 7 | LOW BRAKE AND CLUTCH LUBE PRESSURE | SENGER | (LBCLSF) |
| 1 | 8 | 8 | LOW BRAKE SUPPLY PRESSURE | SENGER | (LBCLSF) |
| 2 | 1 | 17 | LOW CONTROL SYSTEM VOLTAGE | SEVOS | (LCSVSF) |
| 1 | 9 | 9 | LOW ENGINE OIL PRESSURE | SENGER | (LEOPSF) |
| 1 | 10 | 10 | LOW PORT CHARGE PUMP PRESSURE | SENGER | (LPCPSF) |
| 1 | 11 | 11 | LOW PORT HYDRAULIC OIL LEVEL | SENGER | (LPHLSF) |
| 1 | 12 | 12 | LOW SCAVENGE PUMP PRESSURE | SENGER | (LSPPSF) |
| 1 | 13 | 13 | LOW STARBOARD CHARGE PUMP PRESSURE | SENGER | (LSCPSF) |
| 1 | 14 | 14 | LOW STARBOARD HYDRAULIC OIL LEVEL | SENGER | (LSHLSF) |
| 2 | 2 | 18 | LOW VEHICLE SYSTEM VOLTAGE | SEVOS | (LVSVSF) |
| 1 | 2 | 2 | MODE SELECTOR SWITCH FAILURE | SEGMT | (MSSF) |
| 3 | 11 | 43 | PORT AFT PRESSURE TRANS HIGH FAILURE | ANLGIN | (PAMHSF) |
| 3 | 12 | 44 | PORT AFT PRESSURE TRANS LOW FAILURE | ANLGIN | (PAMLSF) |
| 3 | 13 | 45 | PORT BUCKET POTENTIOMETER FAILURE | ANLGIN | (APBCSF) |
| 3 | 14 | 46 | PORT FWD PRESSURE TRANS HIGH FAILURE | ANLGIN | (PFMHSF) |

Table 3.2-2 Error Message Listing (continued)

| Word # | Bit # | Error # | Error Message | Subroutine | Variable Name |
|---|---|---|---|---|---|
| 3 | 15 | 47 | PORT FWD PRESSURE TRANS LOW FAILURE | ANLGIN | (PFMLSF) |
| 4 | 7 | 55 | PORT MOTOR MAG PICKUP FAILURE | FREQIN | (APMSF) |
| 4 | 8 | 56 | PORT SPROCKET MAG PICKUP FAILURE | FREQIN | (APSSF) |
| 2 | 3 | 19 | PORT NO 1 SUSPENSION FAILURE | SEVOS | (P1SCSF) |
| 2 | 4 | 20 | PORT NO 2 SUSPENSION FAILURE | SEVOS | (P2SCSF) |
| 2 | 5 | 21 | PORT NO 3 SUSPENSION FAILURE | SEVOS | (P3SCSF) |
| 2 | 6 | 22 | PORT NO 4 SUSPENSION FAILURE | SEVOS | (P4SCSF) |
| 2 | 7 | 23 | PORT NO 5 SUSPENSION FAILURE | SEVOS | (P5SCSF) |
| 4 | 13 | 62 | PORT WATERJET MAG PICKUP FAILURE | FREQIN | (APWSF) |
| 3 | 1 | 33 | PRIMARY GRILL NOT CLOSED | SESEA | (PGCSSF) |
| 3 | 2 | 34 | RAMP NOT CLOSED AND LATCHED | SESEA | (RCLSSF) |
| 3 | 16 | 48 | STARBOARD AFT PRESSURE TRANS HIGH FAILURE | ANLGIN | (SAMHSF) |
| 4 | 1 | 49 | STARBOARD AFT PRESSURE TRANS LOW FAILURE | ANLGIN | (SAMLSF) |
| 4 | 2 | 50 | STARBOARD BUCKET POTENTIOMETER FAILURE | ANLGIN | (ASBCSF) |
| 4 | 3 | 51 | STARBOARD FWD PRESSURE TRANS HIGH FAILURE | ANLGIN | (SFMHSF) |
| 4 | 4 | 52 | STARBOARD FWD PRESSURE TRANS LOW FAILURE | ANLGIN | (SFMLSF) |
| 4 | 9 | 57 | STARBOARD MOTOR MAG PICKUP FAILURE | FREQIN | (ASMSF) |
| 4 | 10 | 58 | STARBOARD SPROCKET MAG PICKUP FAILURE | FREQIN | (ASSSF) |
| 2 | 9 | 25 | STARBOARD NO 1 SUSPENSION FAILURE | SEVOS | (S1SCSF) |
| 2 | 10 | 26 | STARBOARD NO 2 SUSPENSION FAILURE | SEVOS | (S2SCSF) |
| 2 | 11 | 27 | STARBOARD NO 3 SUSPENSION FAILURE | SEVOS | (S3SCSF) |
| 2 | 12 | 28 | STARBOARD NO 4 SUSPENSION FAILURE | SEVOS | (S4SCSF) |
| 2 | 13 | 29 | STARBOARD NO 5 SUSPENSION FAILURE | SEVOS | (S5SCSF) |
| 4 | 11 | 60 | STARBOARD WATERJET MAG PICKUP FAILURE | FREQIN | (ASWSF) |
| 3 | 3 | 35 | SEAWATER PUMP FAILURE | SESEA | (SEWPSF) |
| 4 | 5 | 53 | STEERING MECHANISM SENSOR FAILURE | AWLGIN | (DTRNSF) |
| 1 | 3 | 3 | TRANSMISSION SELECTOR SWITCH FAILURE | SEGMT | (TSSF) |
| 4 | 12 | 61 | VEHICLE SPEED EXCESSIVE FOR MODE CHANGE | CHOMP | (MLTSSF) |

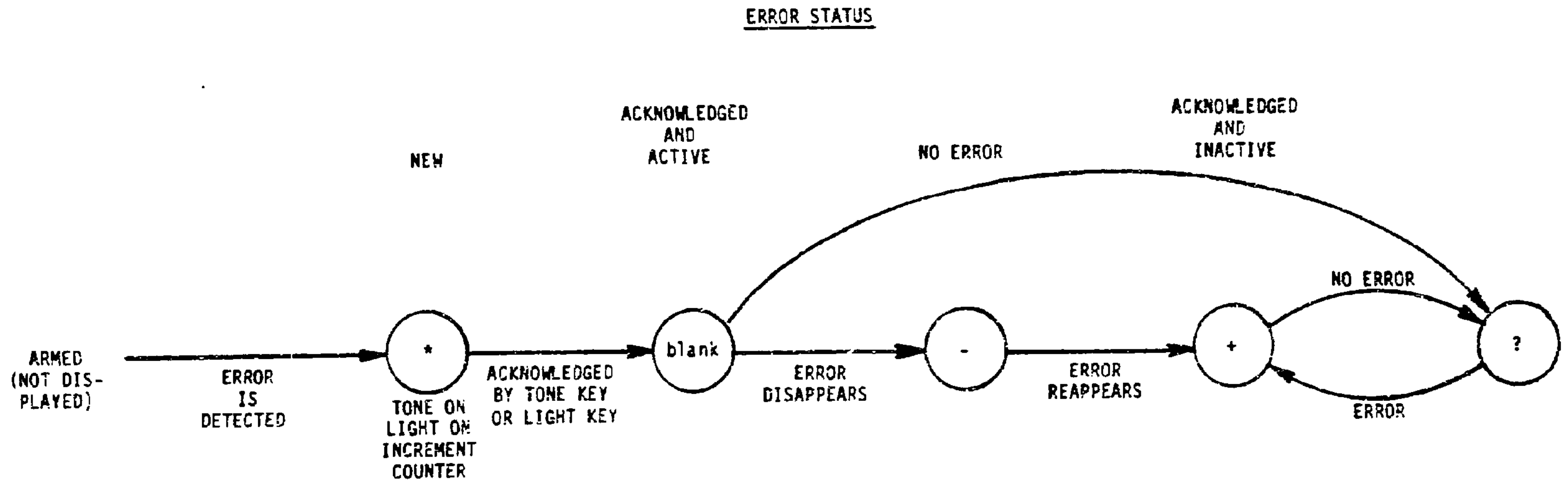
## 4.0 TERRA COMPUTER DESCRIPTION

The purpoe of the Terra computer is to display error messages that are generated by the SC-1. At present, four 16-bit words transfer 60 error messages from the SC-1 via the RS-232 communications port. It is the responsibility of the Terra computer to decode and display and new error messages that have arrived over the communication link. Table 4.0-1 provides a listing of all the error messages for the ATR.

The Terra computer maintains three display screens. They are: the alarm screen, the error log screen, and monitor the screen. The alarm and error log creens are divided into two parts: the top part which displays alarm or log information, and the bottom part which displays communication requests fro the Miltope recorder and allows for the entering of pre-and post-test annotation. The monitor screen is activated by keyboard command.

When error messages are interpreted by the Terra the following sequence of events will occur as shown in Figure 4.0-1.

- A new error will cause the warning light to come on and the warning tone to be activated, and the error message to be displayed in the upper left corner of the screen. All error messages that arrive have an asterisk placed in front of their text until they are acknowledged by the operator by either turning off the warning tone or light. When the errors are acknowledged, a blank space replaces the asterisk. It is possible for an error to become inactive before the operator can acknowledge the error, in that case the error is deemed to be interittent, and a question mark replaces the asterisk.
- Once an error is acknowledged by turning off either the light or the tone, and that error becomes inactive, the blank space is replaced by a minus sign.
- If that error again becomes active the minus sign is replaced by a plus sign.
  - If the error once again becomes inactive the plus sign is replaced with the question mark which means that the error is judged to be unreliable or intermittent.



ERROR DISPLAY

| KEY | ACTION |
|---|---|
| TONE | TONE OFF<br>ALL * TO BLANK |
| LIGHT | LIGHT OFF<br>ALL * TO BLANK |
| REARM | INDIVIDUAL |
| RESET | ALL ALARMS TO ARMED<br>TONE OFF<br>LIGHT OFF |

LOG DISPLAY

| KEY | FUNCTION |
|---|---|
| CONTROL AND CLEAR | SETS ALL COUNTERS TO Ø |
| CONTROL AND STORE | STORES COUNTERS TO BUBBLE MEMORY |
| CONTROL AND ADD | ADDS BUBBLE FILE TO COUNTERS AND RESTORES TO BUBBLE MEMORY |
| CONTROL AND RECALL | RECALLS BUBBLE FILE DATA SCREEN SEND CURRENT SCREEN TO BUFFER |
| CONTROL AND SWITCH | SWITCHES PRIMARY SCREEN AND BUFFER |

Figure 4.0-1 Flowchart Of Error Message Display

- At any point it is possible to rearm (reset) any individual alarm by use of the REARM key (with the cursor pointed to the desired error) or the rearm all alarms by use of the RESET key.
- Any time any alarm (including intermittents) goes from inactive to active the log count for that error increases by one.

The log screens have five function keys associated with them. They are:

CLEAR - Clears all counters to zero.

STORE - Saves the current error log into the stored error log (overwrites it).

ADD - Adds current to stored.

RECALL - Makes the stored error log the current one (overwrites the current error log).

SWITCH - Displays the log screen that is presently not displayed (does not affect either the current or stored error log).

To clarify, there are two screens available for viewing while the vehicle is functioning. They are the alarm screen and the log screen. It is possible to toggle the log screen between the current error display and the stored error display by means of the SWITCH function key. The SWITCH function key does not affect the values in either current or stored error logs.

## 5.0 MILTOPE RECORDER DESCRIPTION

The Miltope recorder used in the ATR vehicle has been reconfigured to provide a parallel communications interfac format with the SC-1 computer. This format has been installed to speed the communications process so that variables can be recorded at .01 second intervals.

Recording operations are initiated by loading a tape into the recorder, turning the recorder power on and simultaneously depressing the CONTROL key and the "R" key. This indicates that a data recording operation is desired and the tape will be rewound and the recorder head will be positioned to the number one track. No previously recorded data can be saved if a used tape is put into the recorder.

The Terra computer then responds by asking for the current date and time. Upon responding, the Terra will ask if any pre-test annotation is desired. If Y is depressed, the Terra will respond by asking what pre-test annotation is desired. If any other key is depressed no pre-test annotation will be possible. If the maximum number of lines of pre-test annotation (24 lines) are entered, or the return key is depressed twice an end to the pre-test annotation will be encountered. The Terra will then respond with the following statement: "To begin data recording activity, hit CONTROL S."

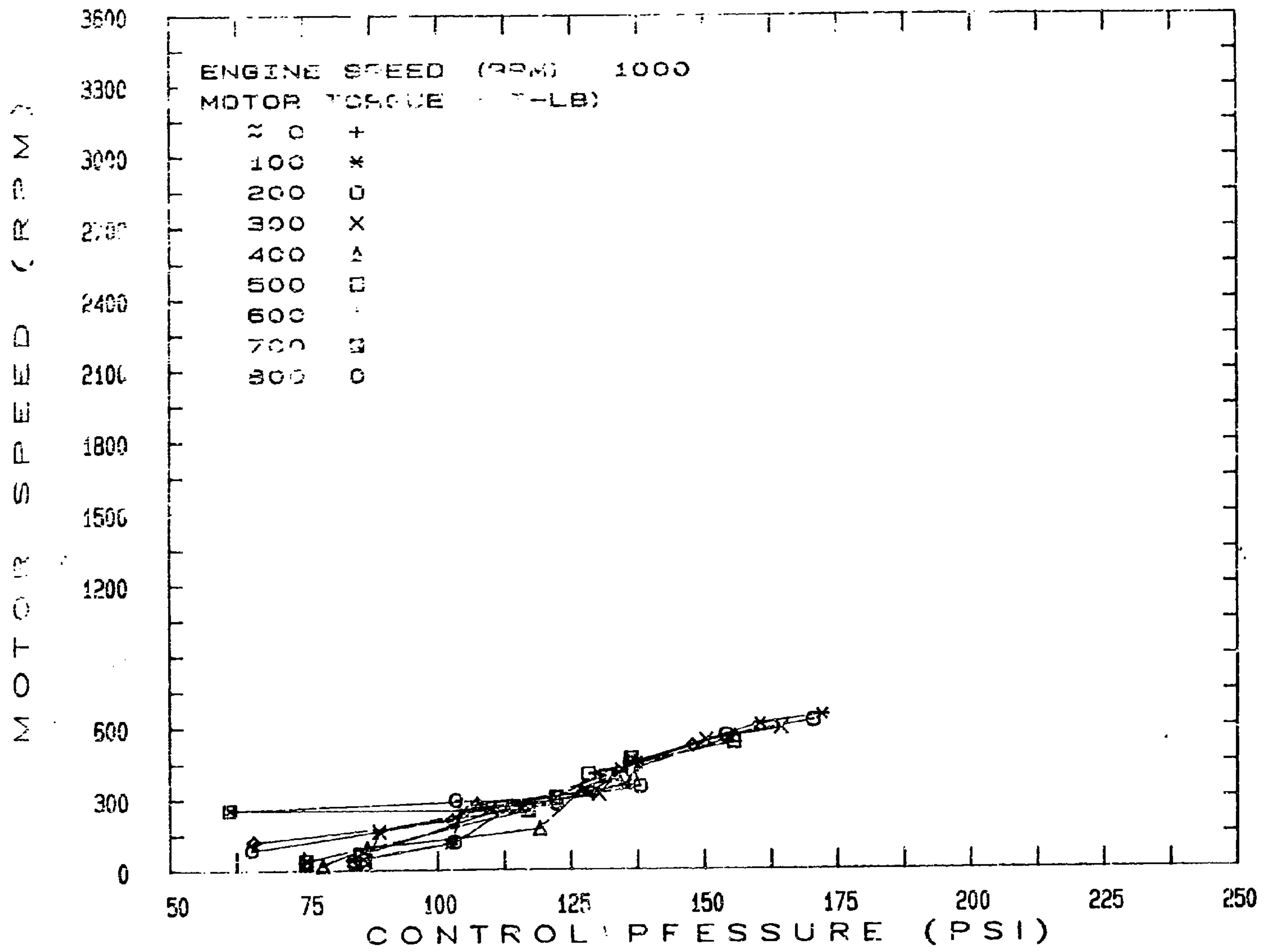
Once the data recording operation is underway, it can be terminated by one of two ways; either CONTROL T can be activated or the tape can come to the end of its track. If the recording of data is terminated before an end of track condition occurs, the Terra will ask if any post-test annotation is desired. If Y is depressed, the Terra will ask what post annotation is desired. Depressing any other key will indicate an end to the data recording activity. If the maximum number of lines of post-test annotation (24 lines) are entered, or the return key is depressed twice, an end to the data recording activity is encountered, the tape recorder will automatically rewind and switch to the next track, if possible. If the maximum number of tracks have been recorded, the Terra will display the following: "Tape full. Replace or depress CONTROL R to write over existing records."

Up to four tracks of data can be recorded on each tape. Each track should contain approximately 20 minutes of data recorded at a rate of 50 variables every .01 seconds.

## 6.0 TRANSMISSION TEST RESULTS

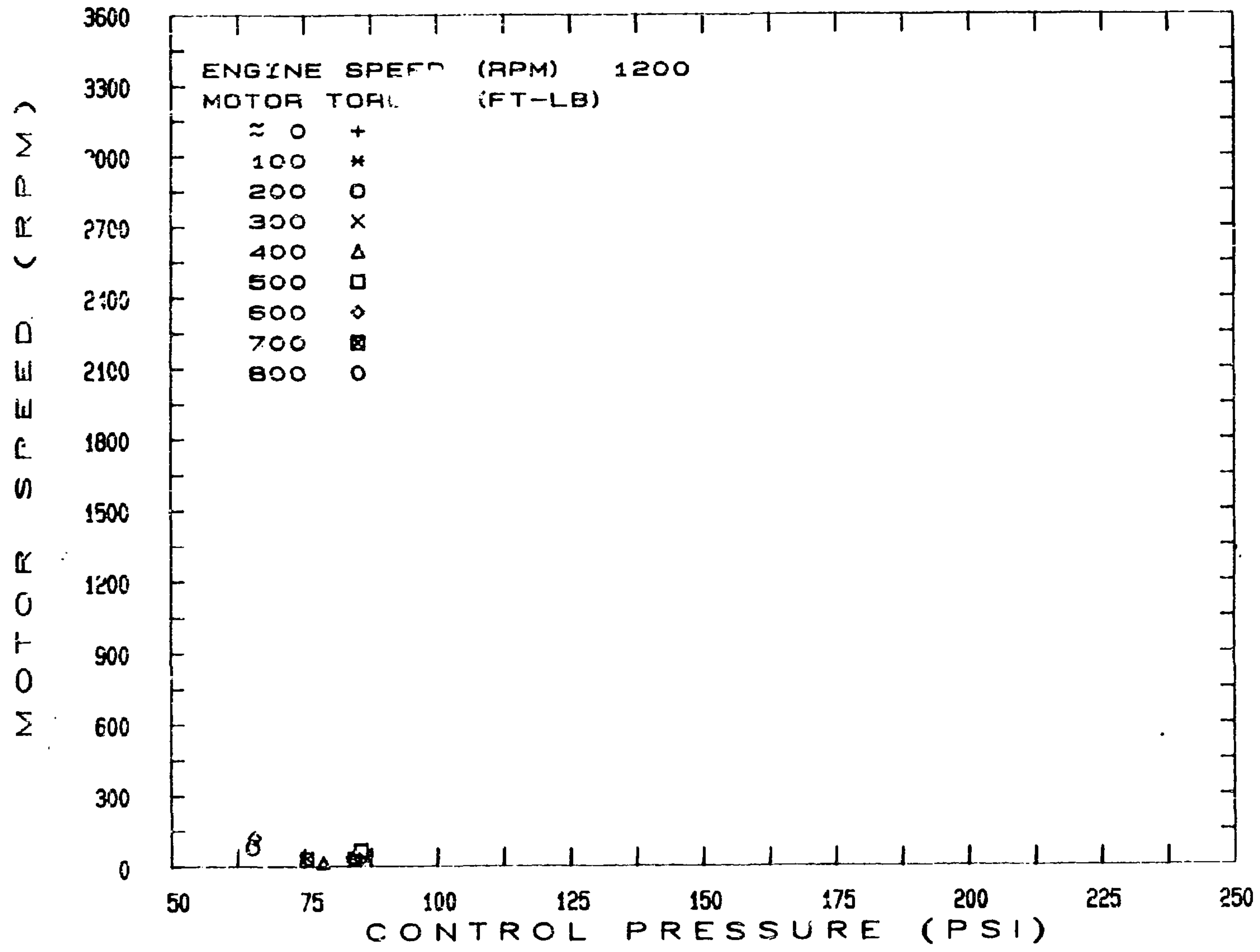
To improve the performance of the transmissions in the ATR vehicle, as compared to the performance obtained in the previously developed Hydrostatic Test Platform, SwRI performed a series of dynamometer tests with Quinto Lubric 822-220 hydraulic fluid. This fluid has superior viscosity over the MIL-H-83282 fire-resistant hydraulic fluid which was previously used.

The test results are presented in graphical form in Figures 6.0-1 through 6.0-10 which show the expected motor speed as a function of control pressure. The predicted efficiency of the transmissions during actual operation is shown in Figures 6.0-11 through 6.0-20. These test results indicate a significant operating improvement of the transmissions using the Quinto Lubric fluid as compared to the MIL-H fire-resistant hydraulic oil. Test results regarding the use of this fluid were presented in a report entitled "Design and Integration of Hydrostatic Transmission in a 300 HP Marine Corps Amphibious Armored Personnel Carrier," prepared under Contract No. N00167-82-C-0156 for the David Taylor Naval Ship Research and Development Center, dated March 1985.



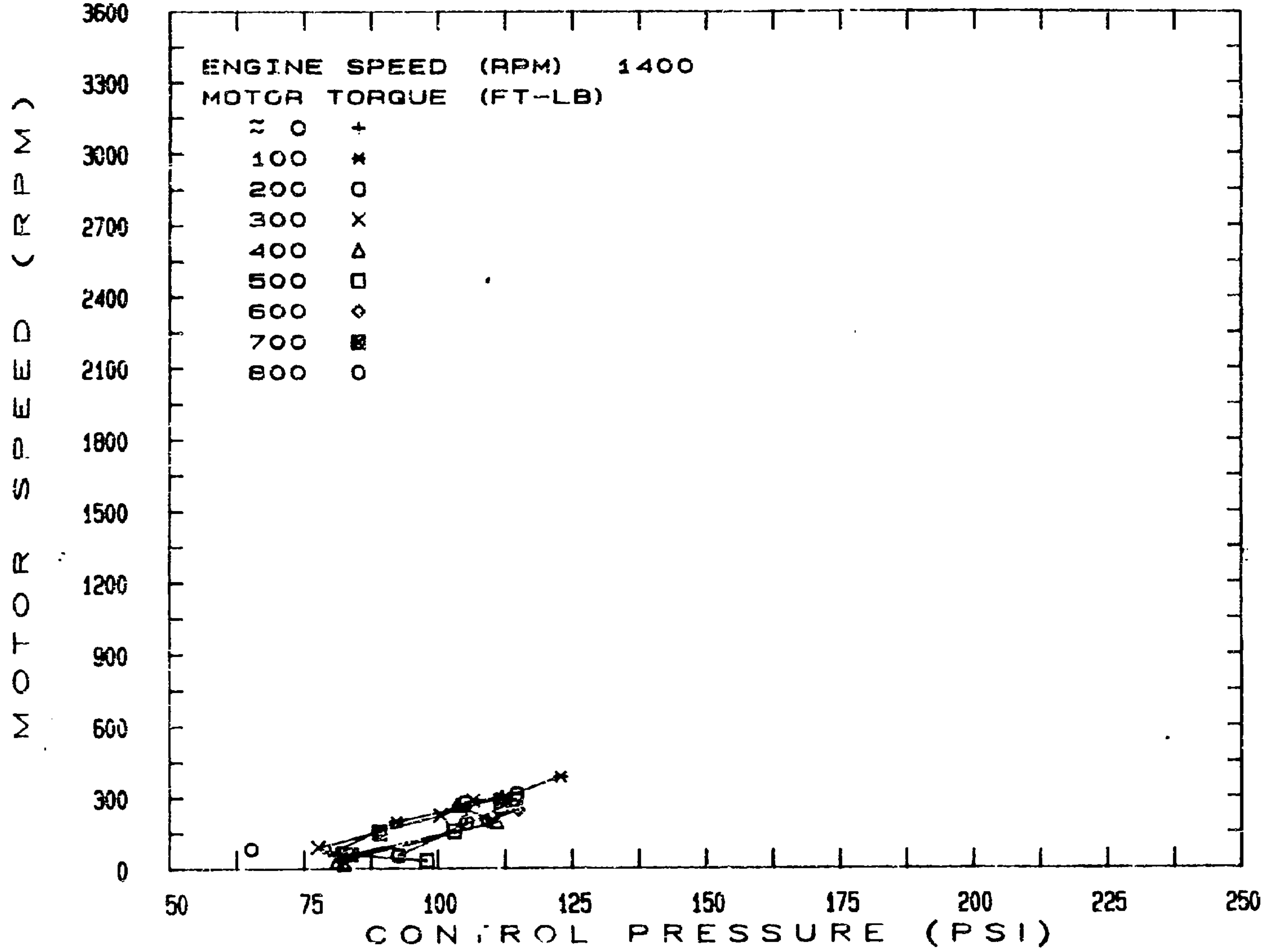
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1000 RPM.

FIGURE 6.0-1



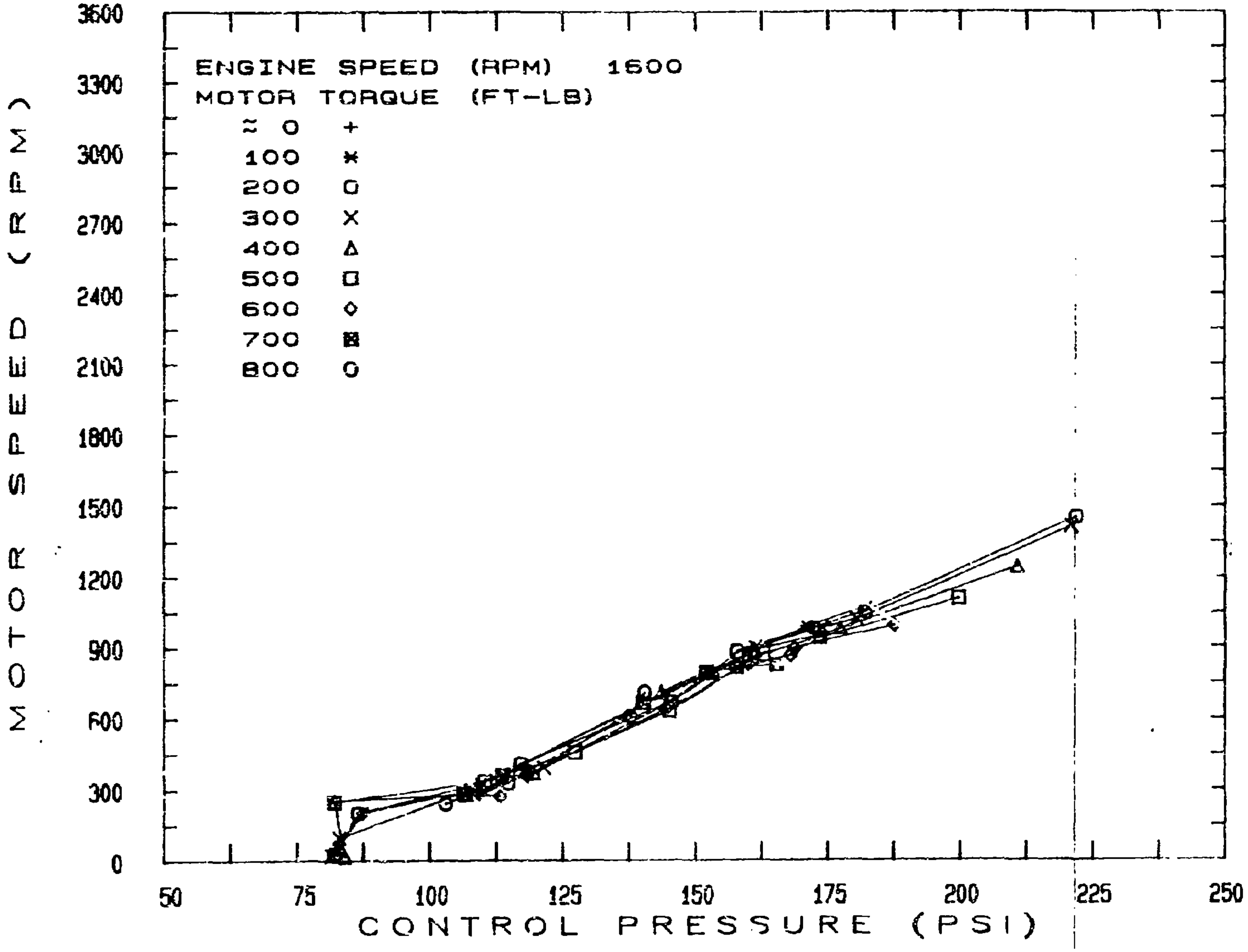
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1200 RPM.

FIGURE 6.0-2



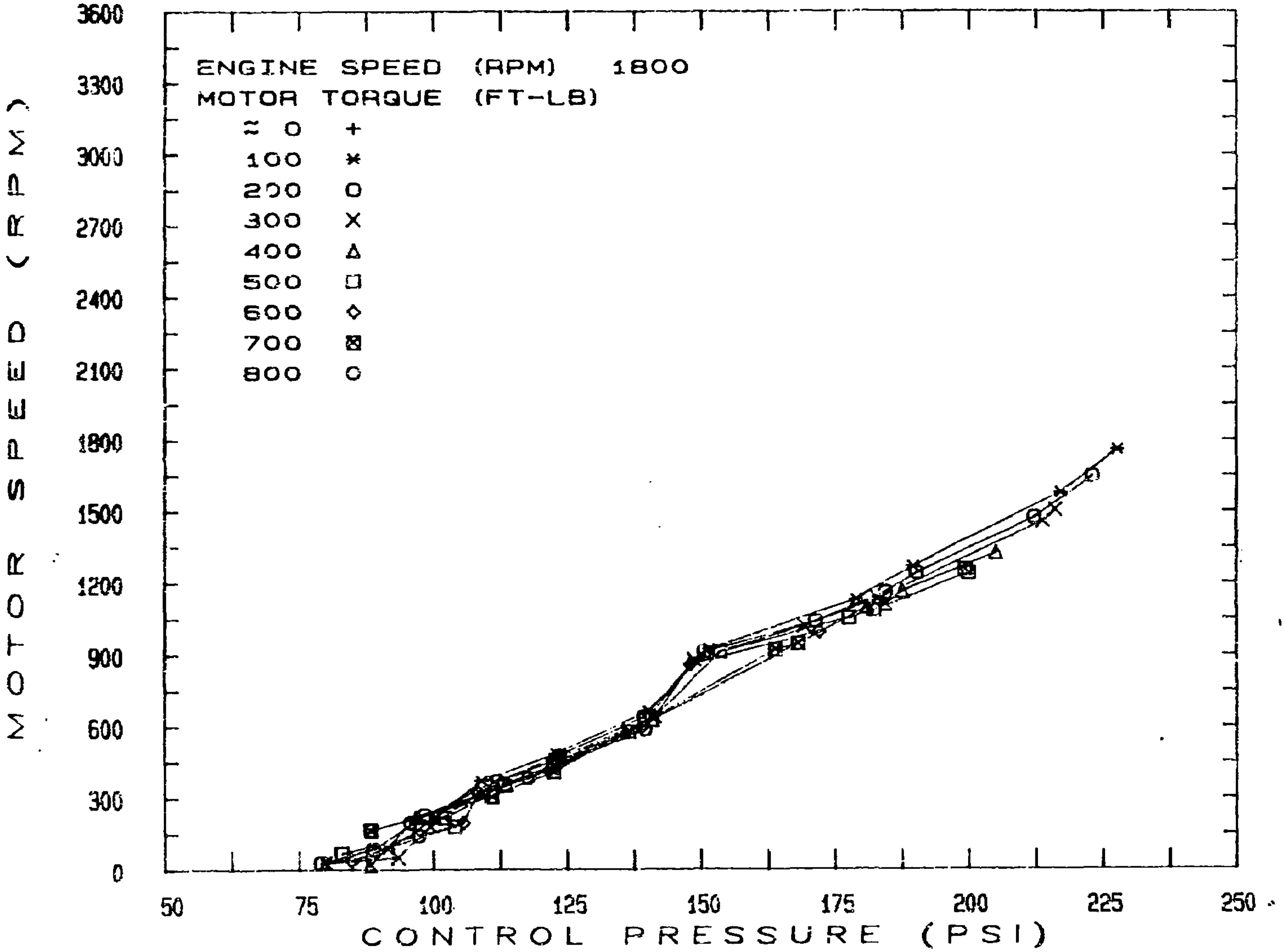
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1400 RPM.

FIGURE 6.0-3



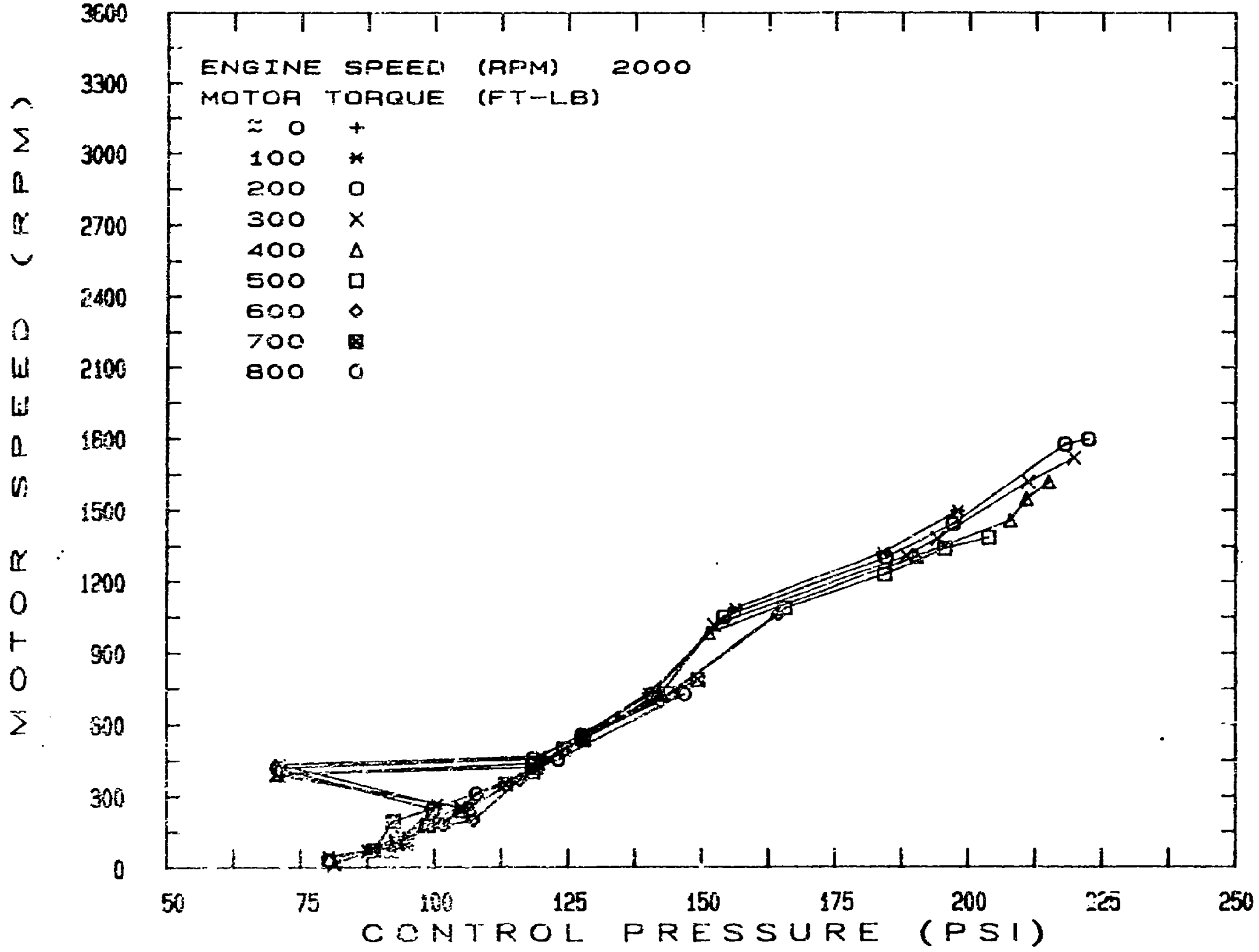
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1600 RPM.

FIGURE 6.0-4



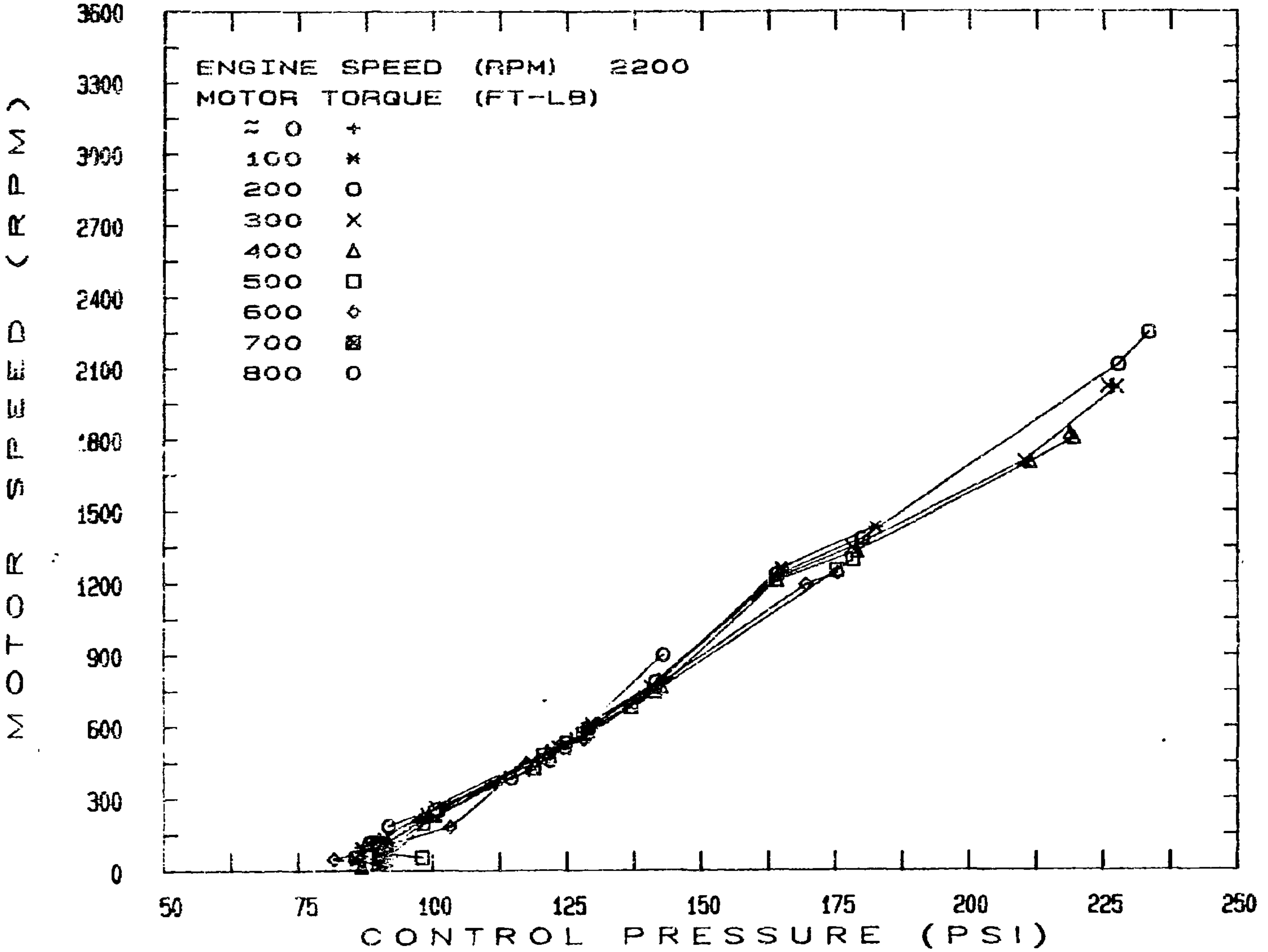
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1800 RPM.

FIGURE 6.0-5



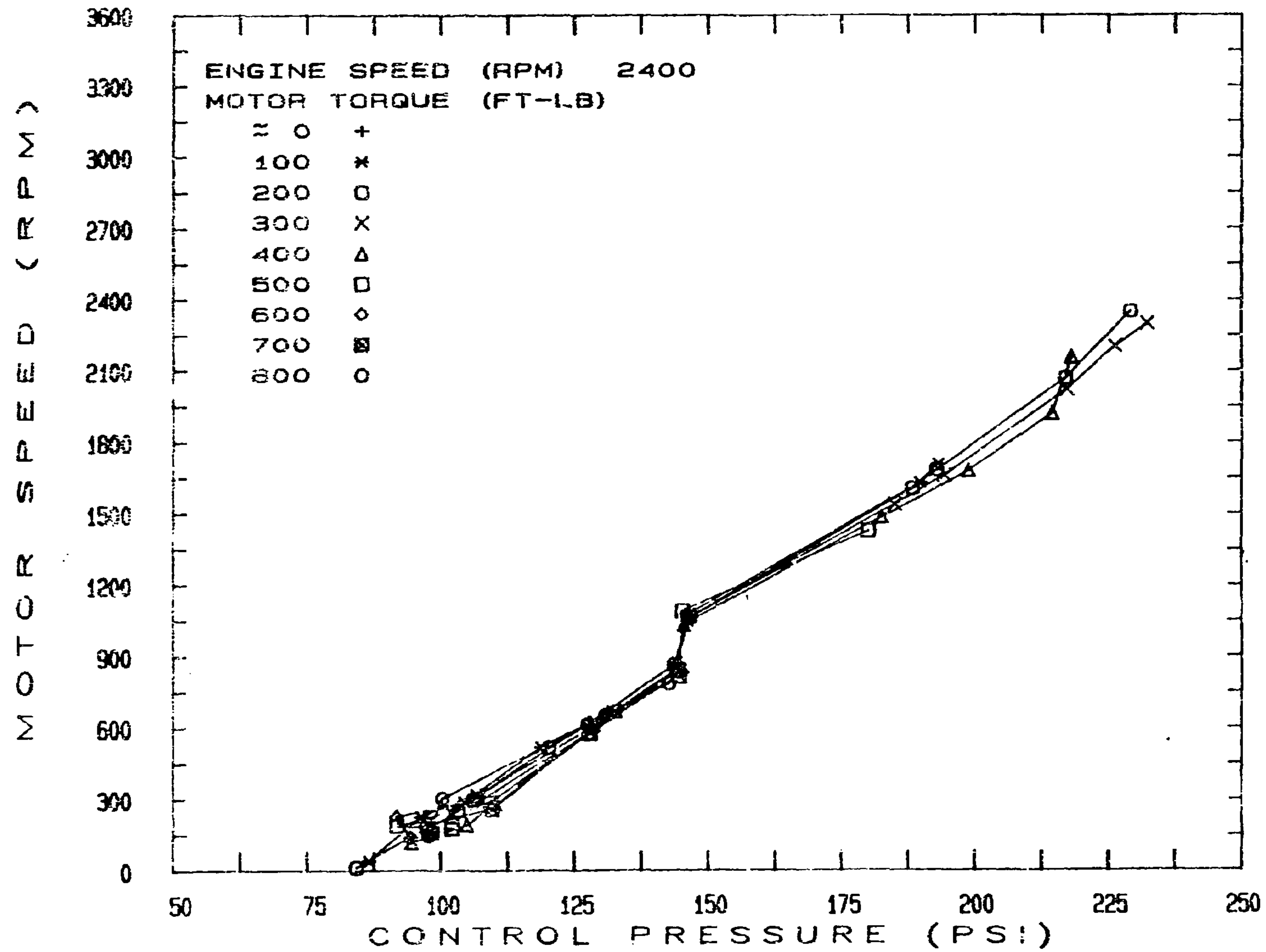
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2000 RPM.

FIGURE 6.0-6



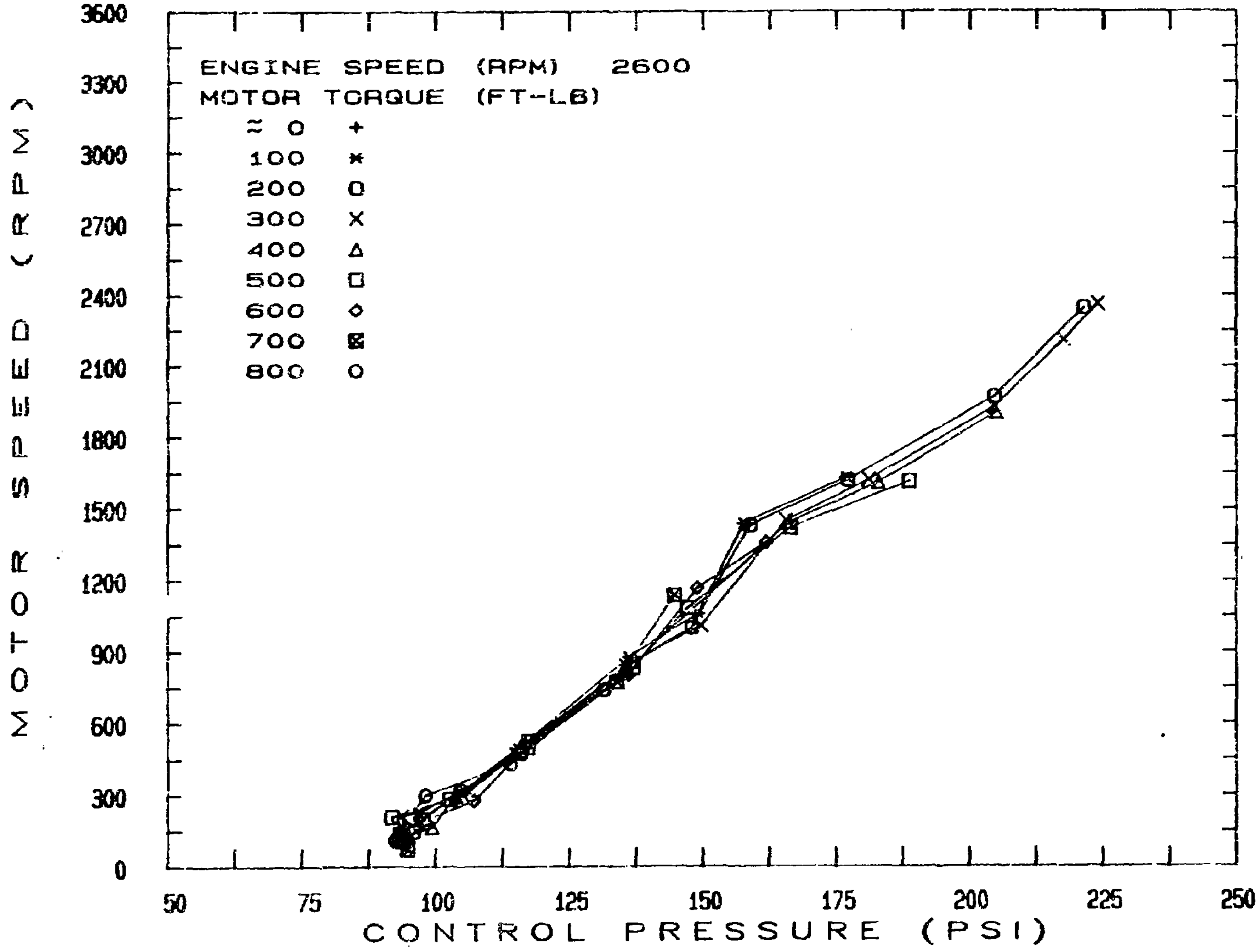
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2200 RPM.

FIGURE 6.0-7



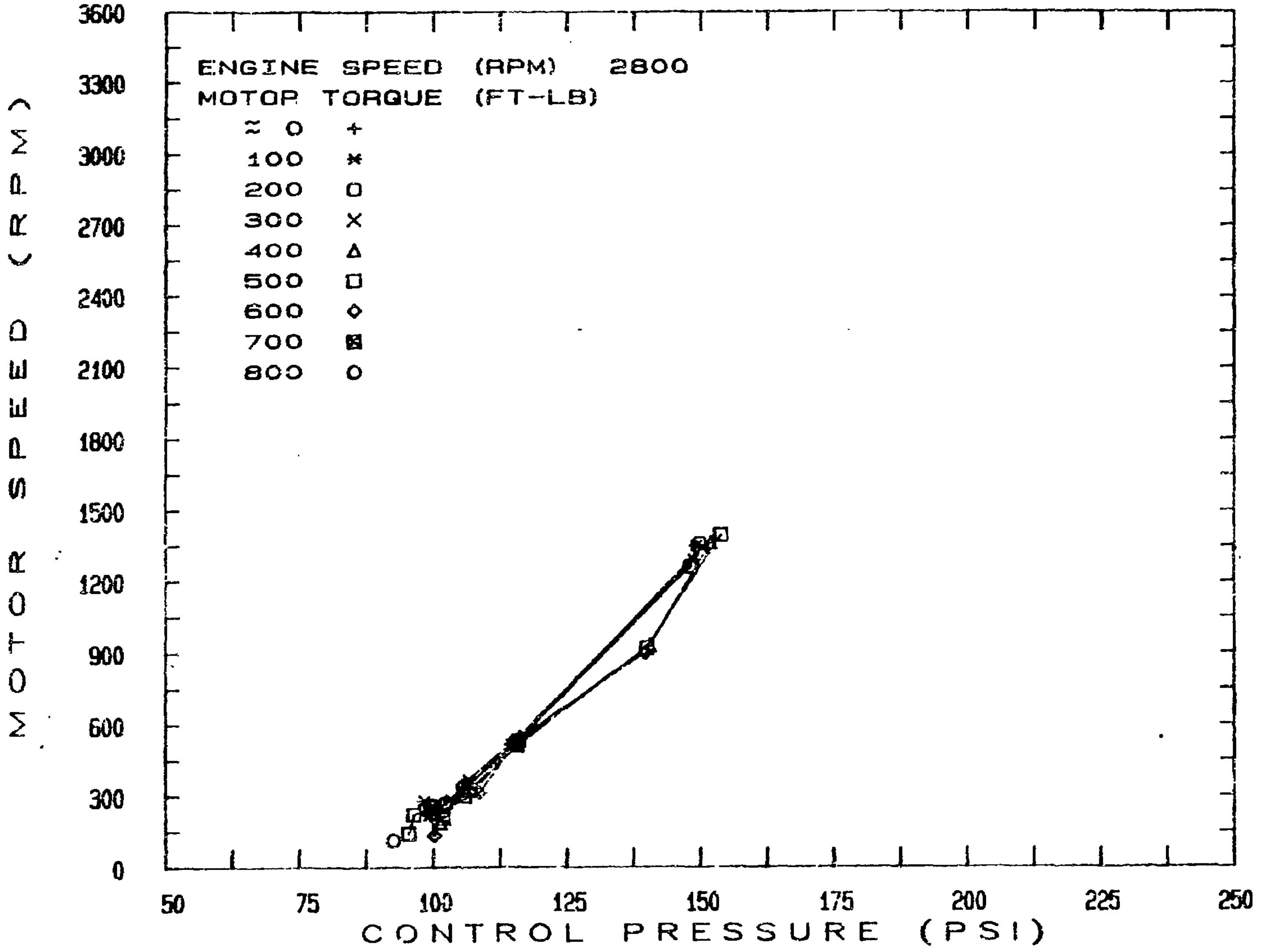
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2400 RPM.

FIGURE 6.0-8



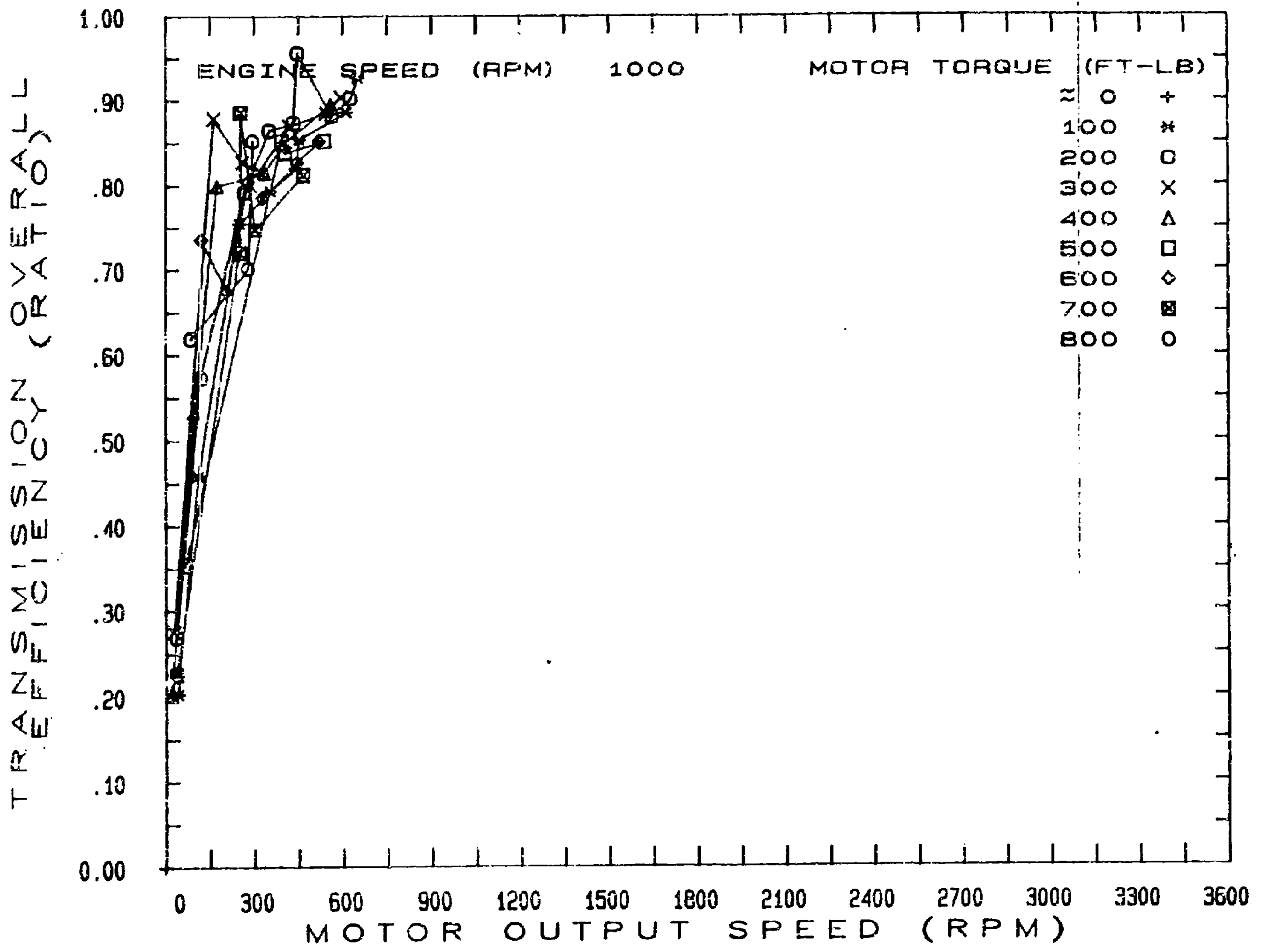
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2600 RPM.

FIGURE 6.0-9



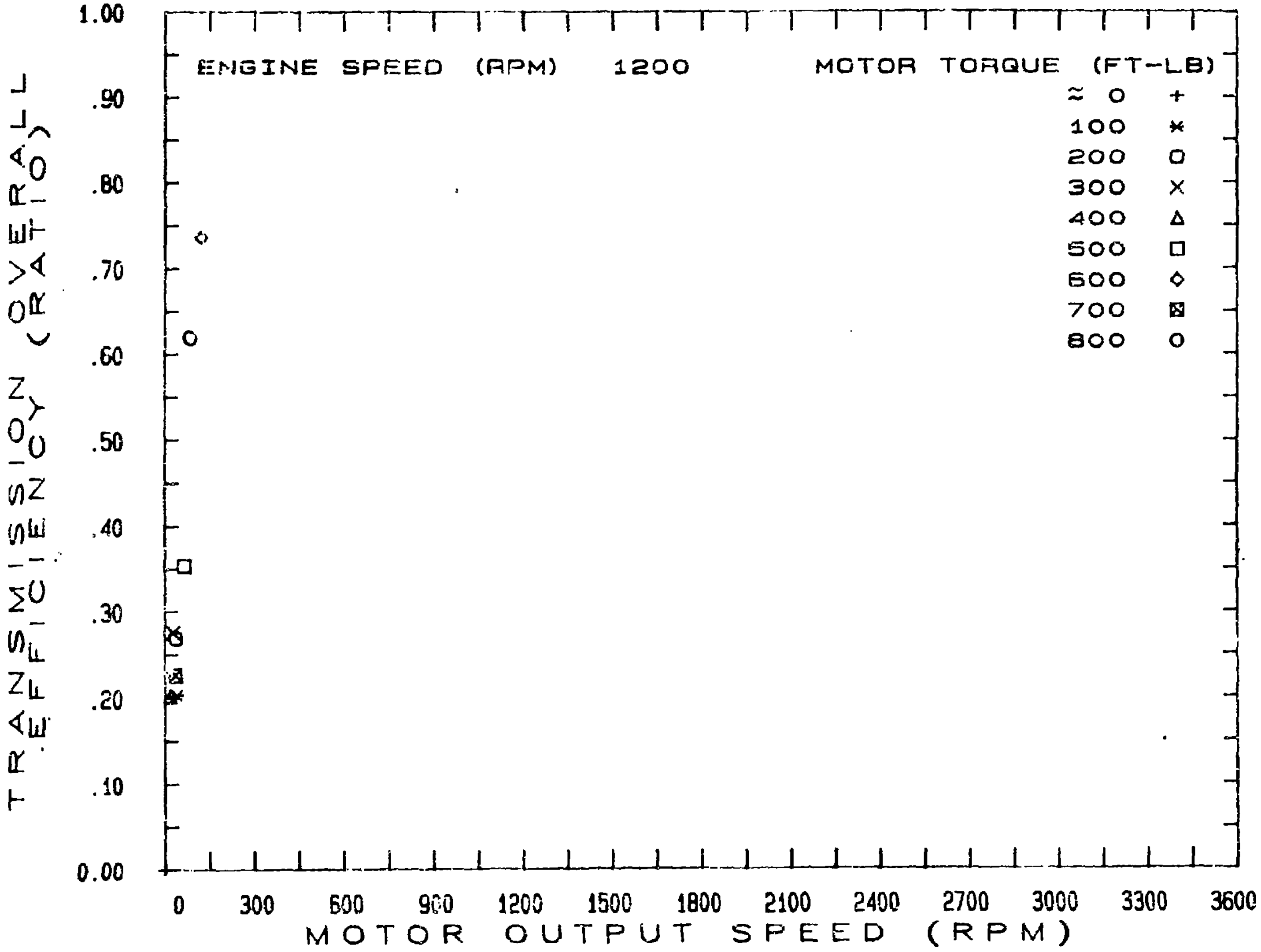
TRANSMISSION MOTOR SPEED VERSUS CONTROL PRESSURE
FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2800 RPM.

FIGURE 6.0-10



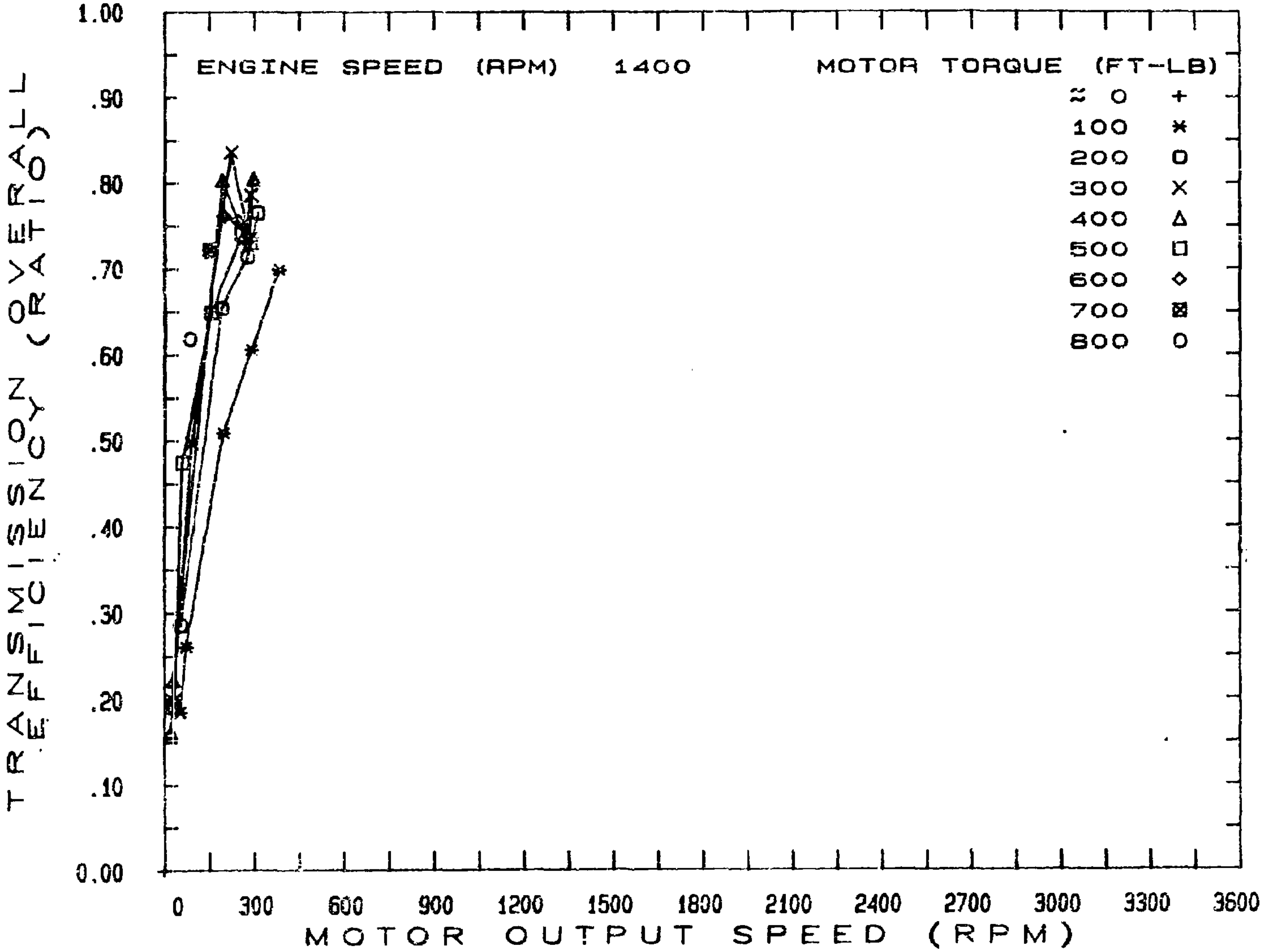
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1000 RPM.

FIGURE 6.0-11



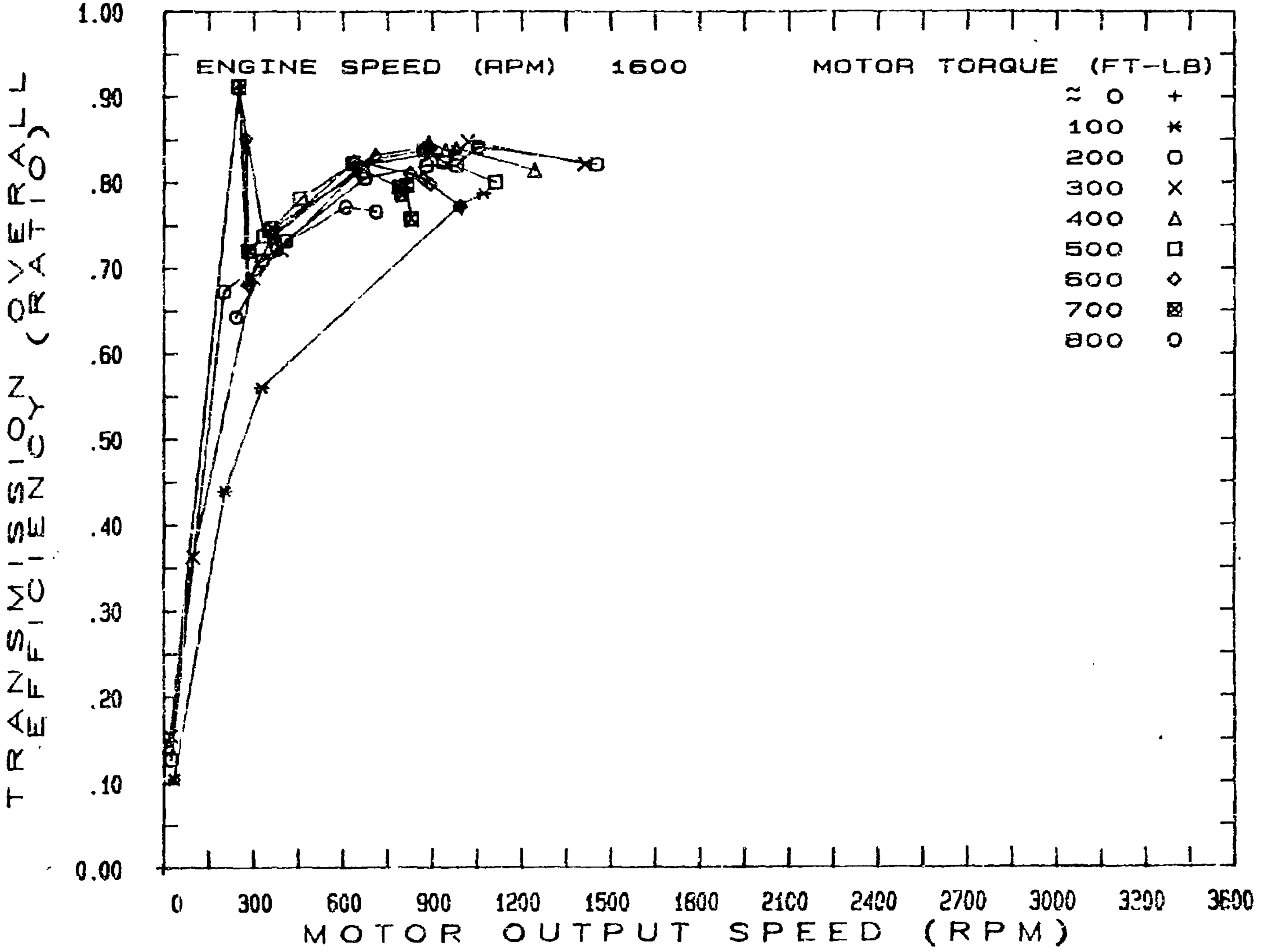
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1200 RPM.

FIGURE 6.0-12



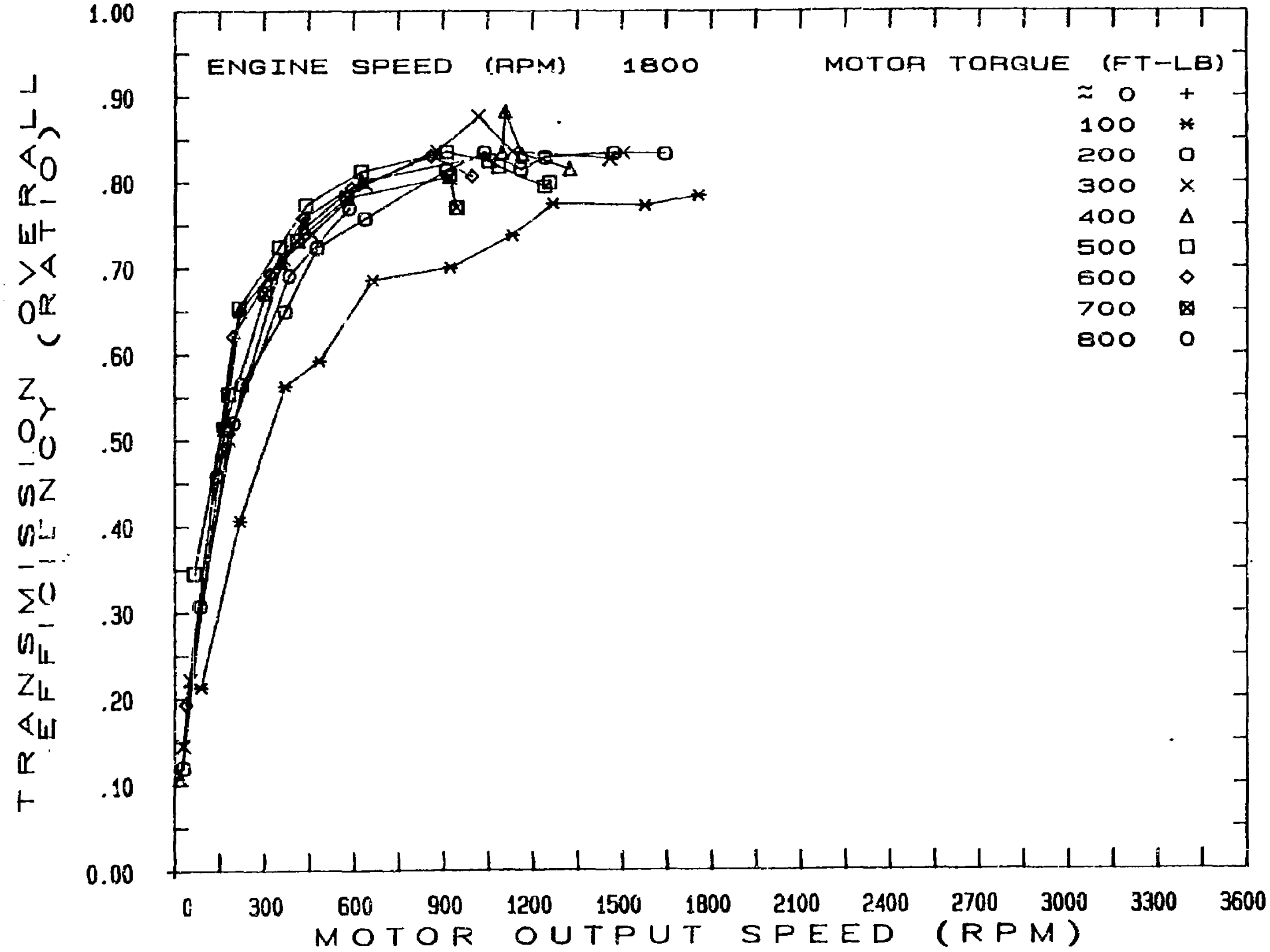
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1400 RPM.

FIGURE 6.0-13



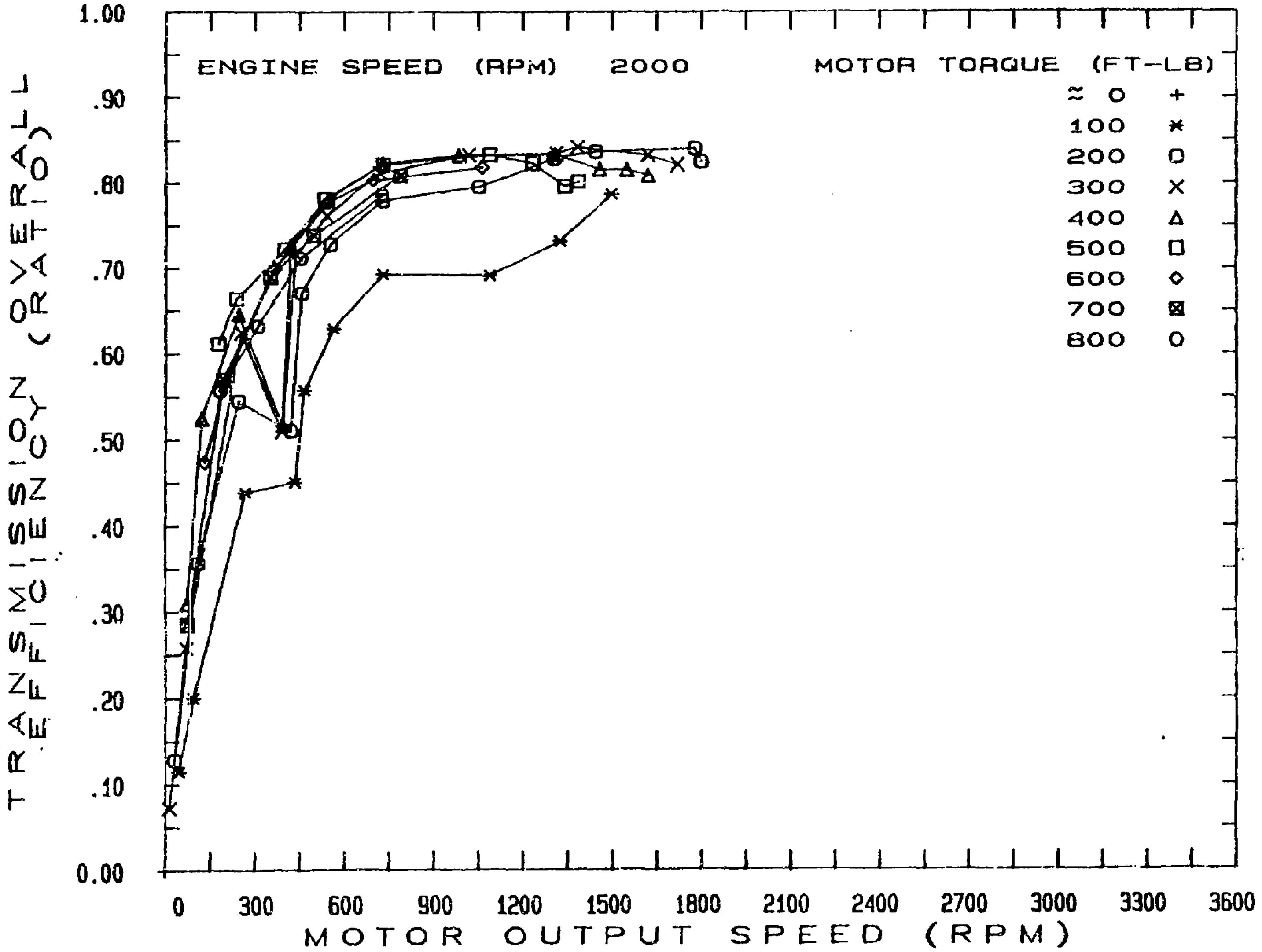
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1600 RPM.

FIGURE 6.0-14



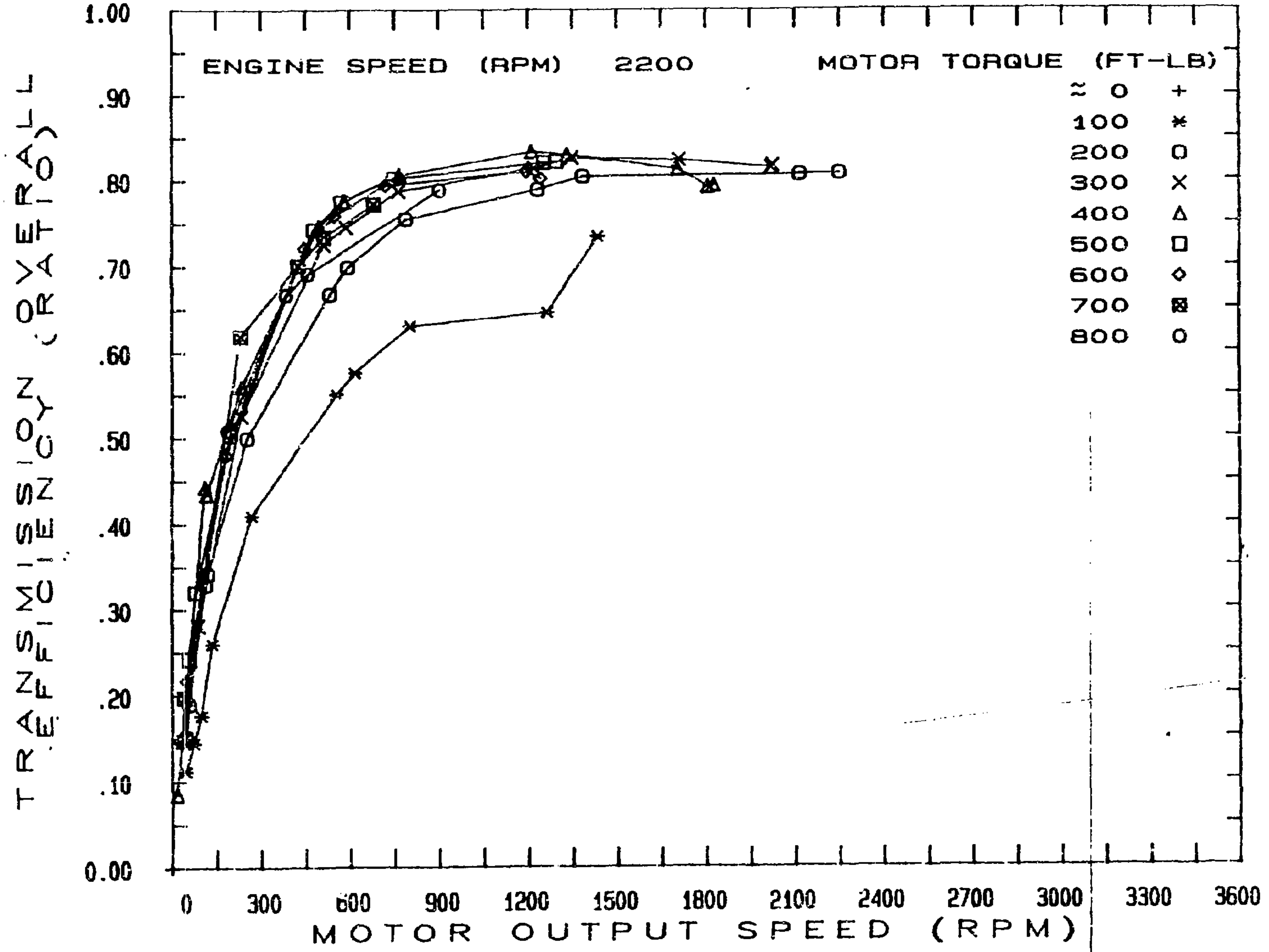
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 1800 RPM.

FIGURE 6.0-15



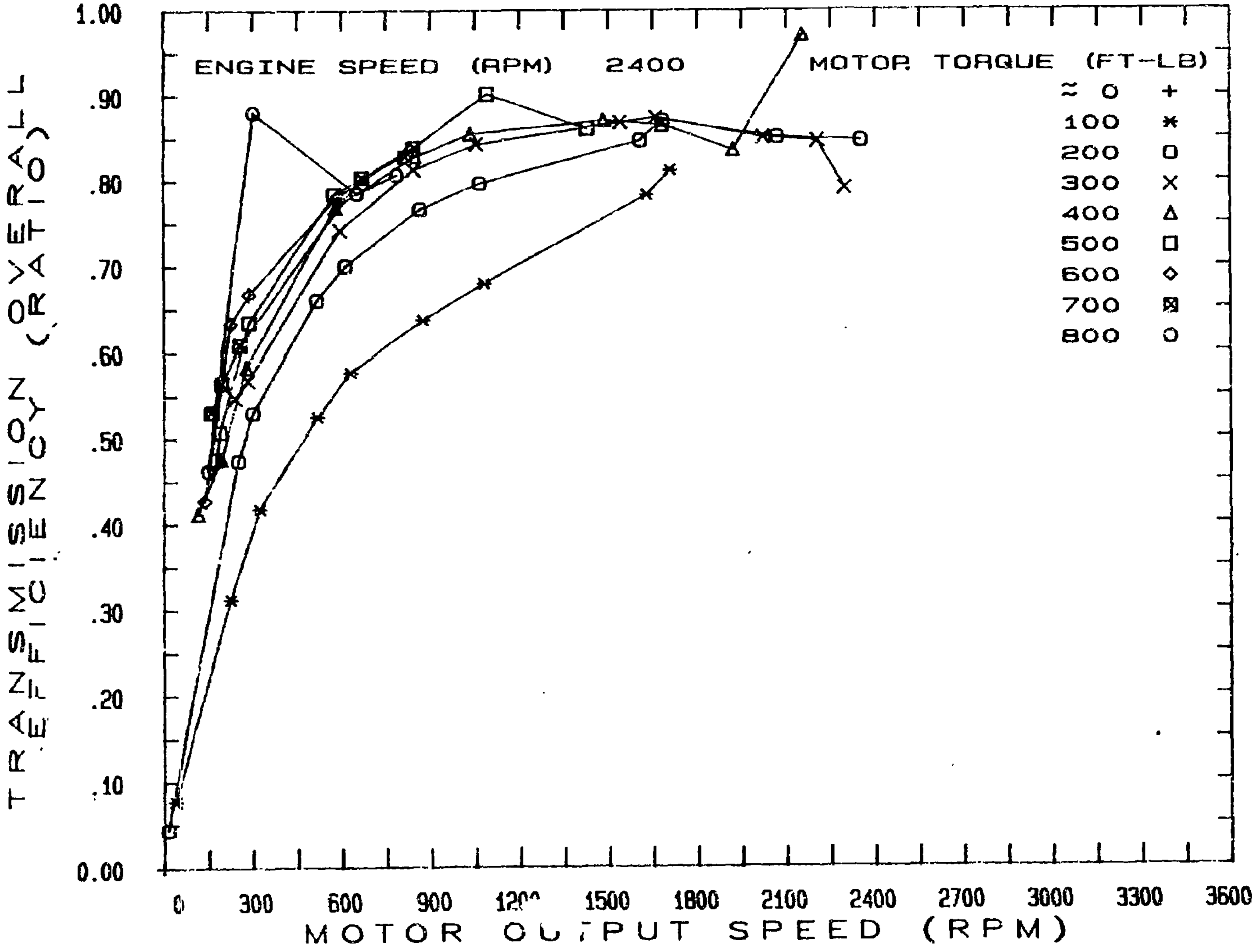
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2000 RPM.

FIGURE 6.0-16



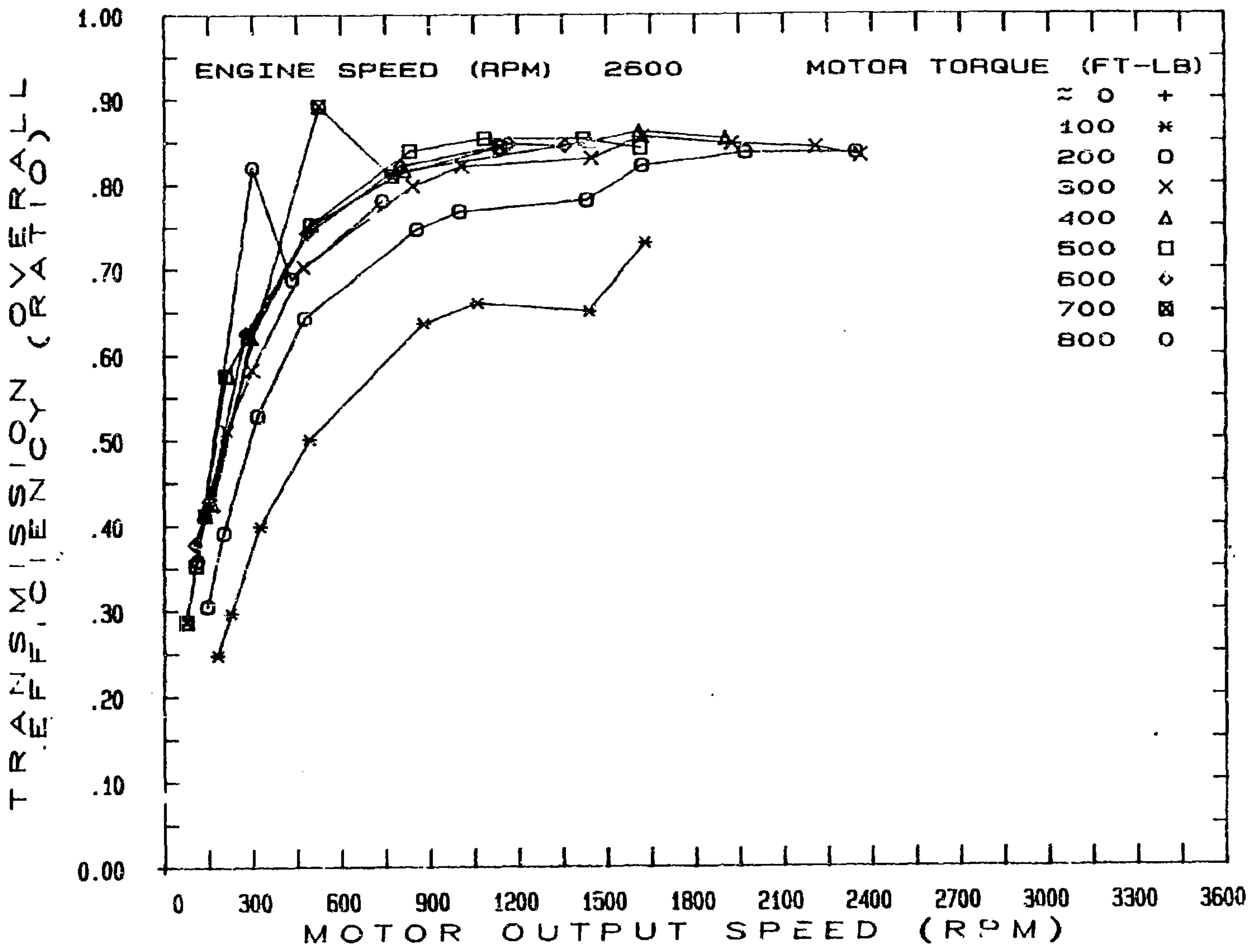
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2200 RPM.

FIGURE 6.0-17



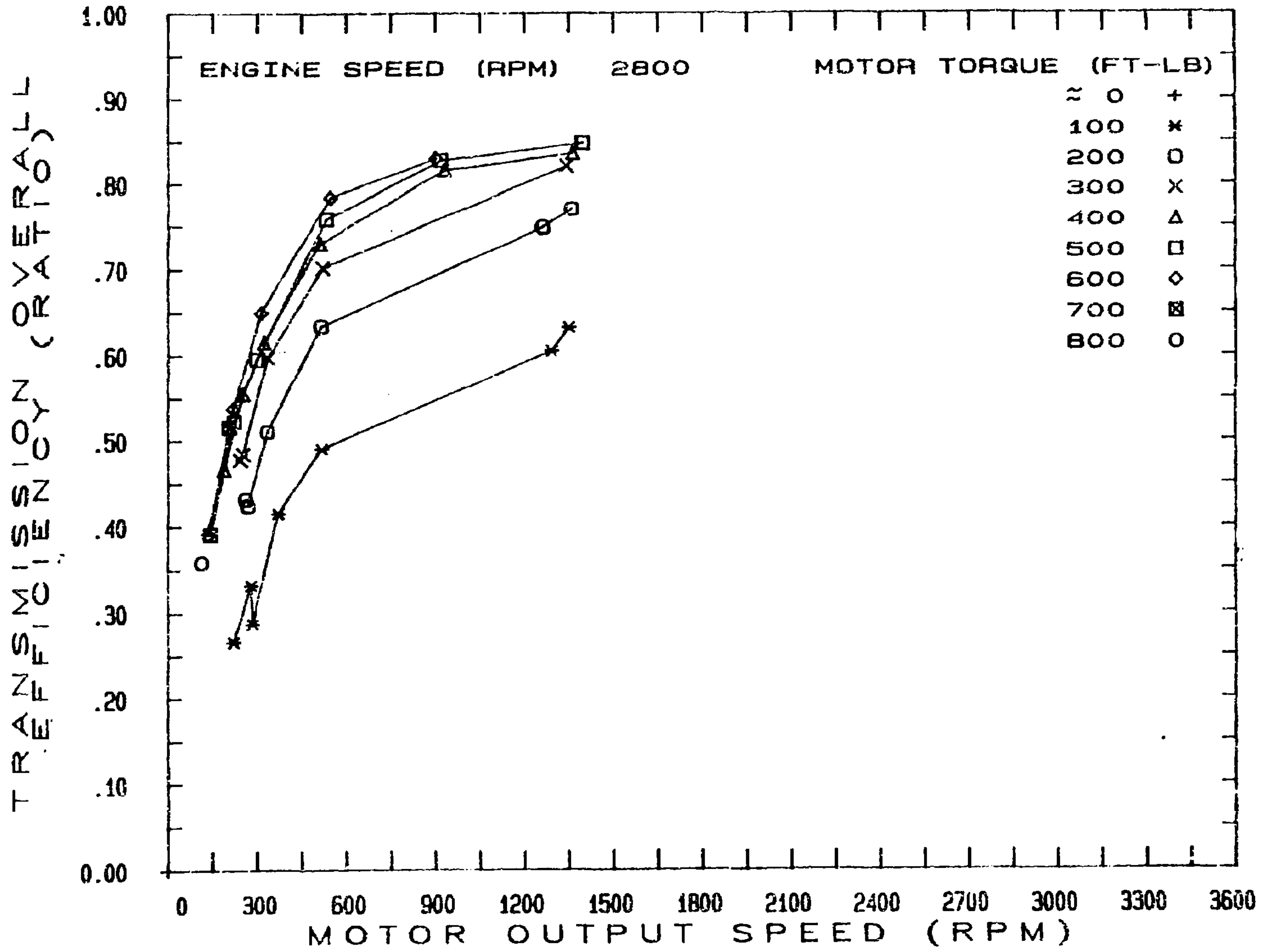
TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2400 RPM.

FIGURE 6.0-18



TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2600 RPM.

FIGURE 6.0-19



TRANSMISSION OVERALL EFFICIENCY VERSUS MOTOR OUTPUT SPEED FOR VARIOUS MOTOR TORQUES AT AN ENGINE SPEED OF 2800 RPM.

FIGURE 6.0-20

## 7.0 TRANSMISSION VALVE DEVELOPMENT

Southwest Research Intitute provided the design and fabrication of a new transmission control valve for use in the ATR vehicle. This valve development effort was undertaken because of operational problems encountered while using the production valve supplied with the transmissions by the manufacturer. The operational problems included an intermittent instability and the inability to perform maintenance on the valve.

Figure 7.0-1 shows a cross section of the valve which was supplied. This valve incorporated the double proportional valve which was provided by the original manufacturer, but also incorporates a flip-flop valve and a shuttle valve in the same overall design. Figure 7.0-2 shows the relationship between pressure and control voltage that was obtained from this valve.

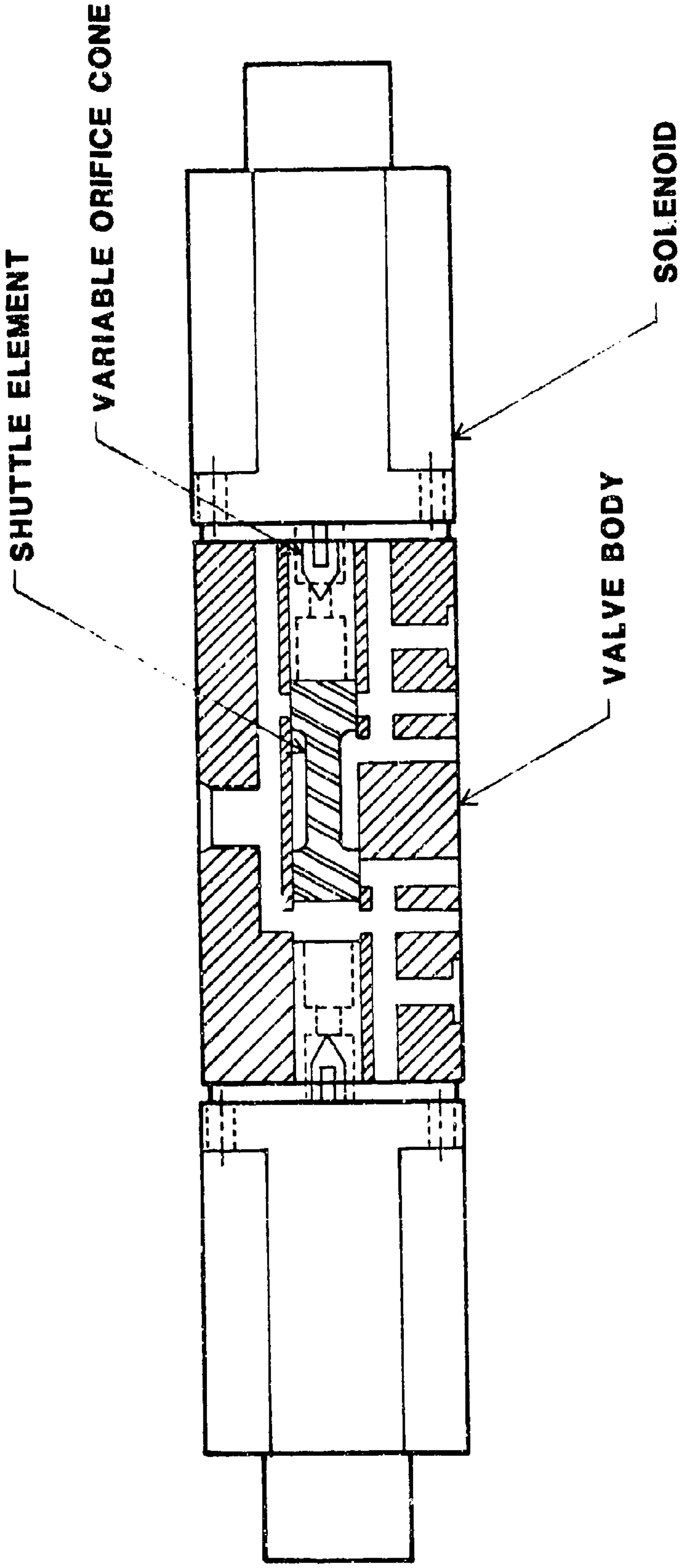


Figure 7.0-1 Transmission Valve Section View

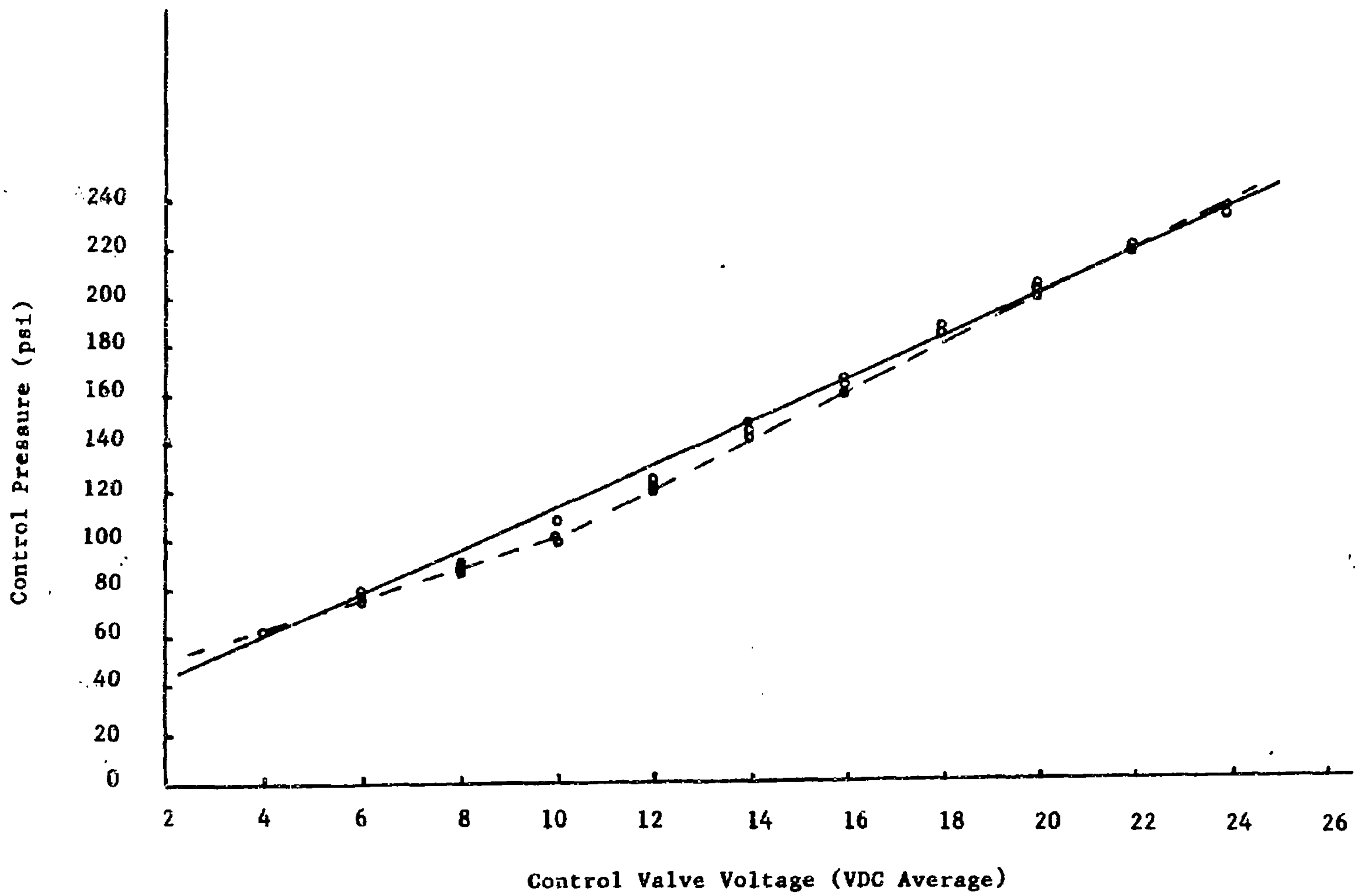


Figure 7.0-2 Control Pressure Versus Control Valve Voltage

# Appendix A

## ATR Control Software Listing

```
SERIES-III FORTRAN-86 COMPILER V1.0
 MPILER INVOKED BY: :F1:FORT86.86  :F2:ATR.FOR  CODE TITLE(13:25:00 11 DEC.85)

   1           PROGRAM ATR
        C*****************************************************************
        C*****************************************************************
        C*****                                                      ******
        C*****            PROJECT:         AUTOMOTIVE TEST RIG      ******
        C*****            SPONSORS:        AAI CORP./DTNSRCDC       ******
        C*****            CONTRACT #:      03-8284-001              ******
        C*****            PROGRAMER:       BENJAMIN A. TREICHEL SwRI ******
        C*****            DATE/VERSION:    DEC. 08 1985/V2.1        ******
        C*****                                                      ******
        C*****************************************************************
        C*****************************************************************
     =1 $      INCLUDE (:F2:COMMON.FOR)
    2=1        INTEGER*4 RESULT,MASK(16),NAME
    3=1        INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
    4=1        INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
    5=1        INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
    6=1        INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
    7=1        INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
    8=1        INTEGER*4 BIT19,BIT29,BIT39,BIT49
    9=1        INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
   10=1        INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
   11=1        INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
   12=1        INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
   13=1        INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
   14=1        INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
   15=1        INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   16=1        INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   17=1        INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   18=1        INTEGER*4 DG26,DG27,DG28,DG29,DG30
   19=1        INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
   20=1        LOGICAL*4 MSSF,TSSF,BSSF,BKNFSF,FGSCSF,INDFSF
   21=1        LOGICAL*4 HEOTSF,LSCLSF,LSRPSF,LSSPSF,LPCPSF
   22=1        LOGICAL*4 LSCPSF,MFBPSF,LEOPSF,LPHLSF,LSHLSF
   23=1        LOGICAL*4 TBIT,LSPPSF,F1RESF
   24=1        LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
   25=1        LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
   26=1        LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
   27=1        LOGICAL*4 AEBPSF,SEWPSF,HBPVSF,FEBPSF,PGCSSF
   28=1        LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
   29=1        LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
   30=1        LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
   31=1        LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
   32=1        LOGICAL*4 MOOFCC,MLTSSF,RENTRY
   33=1        LOGICAL*4 NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
   34=1        LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
   35=1        INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
   36=1        INTEGER*4 IAPBCN,IASBCN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
   37=1        INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
   38=1        INTEGER*4 TRNDIR
   39=1        INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
   40=1        INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
   41=1        INTEGER*4 TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG
   42=1        INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
   43=1        INTEGER*4 CKPSLT,CKRSL1,CKRSL2
   44=1          INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
   45=1          INTEGER*4 DTRST,PRMOOP
   46=1          INTEGER*4 TDELAY,TIMER,DMOOP,PDMOOP,DIGOUT(8)
   47=1          INTEGER*4 SECPOP,PCPWNP,BLOPON
   48=1          INTEGER*4 OCTD,ORCLSE,OOTD,OROPEN,PCPWBP,BLOOFF,SLOWER
   49=1          INTEGER*4 SUDOWN,BLCVV,BHCVV,MKSPON,MSPON,TRKSTP,SURISE
   50=1          INTEGER*4 ERRDLY,DTDOS,PDTDOS,PTDOS
   51=1          INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
   52=1          INTEGER*4 RSMS,RPMS,RSSS,RPSS,REJ
   53=1          INTEGER*4 PTVV,STVV
   54=1          REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
   55=1          REAL*4 FDR,WJCNST,INTCPT( 1:1)
   56=1          REAL*4 PPMTPR,SPMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1          REAL*4 DPMSP,DSMSP,OBHMSP,PNT18,AUX9,AUX10
   58=1          REAL*4 BHWSP,BHWSR,BPWSP,BSWSP
   59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFF
   60=1          REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
   61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFF
   62=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
   63=1          REAL*4 DPPRT,PDIFF,DPPWJ,DSPRT,SDIFF,DPSWJ
   64=1          REAL*4 PPDIS,PPDFF,APSSP,ASSSP
   65=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
   66=1          REAL*4 SPDIS,SPDFF,MAXMSP,TREP,DES
   67=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1          REAL*4 M29,M30,M31,M32,M33,M34
   74=1          REAL*4 DHMSPD(4),PAMPR(10),SAMPR(10),PPMPR(10),SPMPR(10)
   75=1          INTEGER*4 M2,M3,M10,M11,M12
   76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1          COMMON /BITFNC/ RESULT,MASK,NAME
   79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1          COMMON /ERROR/MSSP,TSSP,OSSP,BKNPSP,PSSCSP,INSPSP
   98=1          COMMON /ERROR/ HEOTSP,LECLSP,LBRPSP,LESPSP,LPCPSP
   99=1          COMMON /ERROR/ LSCPSP,HPBPSP,LEOPSP,LPHLSP,LSHLSP
  100=1          COMMON /ERROR/ LSPPSP,FIRESP
  101=1          COMMON /ERROR/ P1SCSP,P2SCSP,P3SCSP,P4SCSP,P5SCSP
```

```
      COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
      COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
      COMMON /ERROR/ AEBPSF,SEWPSF,HDPVSF,FEBPSF,PGCSSF
      COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INITIAL,SFTINP
      COMMON /ERROR/ APBCSF,ASBCSF,DAMHSF,DAMLSF,DAMRSF,DTRNSF
      COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
      COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
      COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
      COMMON /ERROR/ NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
      COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
      COMMON /CINOUT/ IAPBCN,IASBCN,IDAMSP,IDAMSR,IPAMPR,IPFMPR
      COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
      COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
      COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
      COMMON /CINOUT/ TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG
      COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
      COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
      COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
      COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
      COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
      COMMON /CINOUT/ SECFOF,PCFWNP,BLGPON
      COMMON /CINOUT/ GCTD,GRCLSE,GOTD,GROPEN,PCFWBP,BLGOFF,S' UWER
      COMMON /CINOUT/ SUDOWN,DLCVV,DHCVV,MNBPON,MBPON,TRKSTP,SORTSE
      COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
      COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
      COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
      COMMON /CINOUT/ PTVV,STVV
      COMMON /CALC/ APBCAN,ASBCAN,DAMSP,DAMSR,DTRNR,PAMTPP
      COMMON /CALC/ FDR,WJCNST
      COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
      COMMON /CALC/ DPMSP,DSMSP,ODAMSP,PNTIG,AUX9,AUX10
      COMMON /CALC/ DAWSP,DAWSR,DPWSP,DSWSP,INTCPT
      COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
      COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
      COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
      COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
      COMMON /CALC/ DPFRT,PDIFP,DPFWJ,DSFRT,SDIFP,DPSWJ
      COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
      COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
      COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
      COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
      COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
      COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
      COMMON /CALC/ DAMSPB,PAMPR,SAMPR,PFMPR,SFMPR
      COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
      COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
      COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
      COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
      COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
C
      CALL INIT
      CALL START(IAPBCN)
   10 CONTINUE
      DO 20 I=1,8
      DIGOUT(I)=0
   20 CONTINUE
      DO 30 I=0,15
      MASK(I+1)=2**I
      30      CONTINUE
              I=0
              M1=700.1
      C
      C       INITIALIZATION
      C
              INTIAL=.TRUE.
              DIGOUT(3)=HYBPSS
              CALL ERROUT
              CALL SYNC1
              CALL SYNC2
      40      CONTINUE
              CALL SYNC1
              ENGNON=SBITS(DIGIN(JCARDS),WIDTHS,BEGINS)
              LPGPSF=TBIT(ENGNON,BIT5)
              LSGPSF=TBIT(ENGNON,BIT6)
              DIGOUT(7)=ENGMSG
              AENSP=FLOAT(600*IAENSP)/134.0
              IF(AENSP.LT.400.0) THEN
                  CALL SYNC2
                  GOTO 40
              ENDIF
              PDTDOS=LOW
              PRMODP=LAND
              DMODP=TRNSTN
              DIGOUT(3)=HYNBPS
              DIGOUT(7)=SUSMSG
              DIGOUT(1)=SLOWER
              ITIMER=0
      50      CALL SYNC1
              IF(ITIMER.NE.300) THEN
              ITIMER=ITIMER+1
              IF(TDELAY.NE.0) THEN
                  TIMER=TIMER+1
                  IF(TIMER.GE.TDELAY) CALL CHOMP
                  GOTO 60
              ENDIF
      C
      C       ARE WE PRESENTLY INVOLVED IN A MODE CHANGE ?
      C
              IF(.NOT.MODPCC) THEN
                  CALL CHOMP
                  GOTO 60
              ENDIF
      C
      C       IS A MODE CHANGE DESIRED ?
      C
              IF(PRMODP.NE.DMODP) THEN
                  PDMODP=DMODP
                  MODPCC=.FALSE.
                  CALL CHOMP
              ENDIF
      C
      60      CONTINUE
              CALL SYNC2
              GOTO 50
              ENDIF
              INTIAL=.FALSE.
```

```
 207            DIGOUT(7)=NSUSMG
 208            TIMER=0
          C
          C
          C     SYNCHRONIZE WITH HARDWARE BEFORE START OF MAJOR TIMING LOOP
          C
 209      100   CONTINUE
 210            CALL SYNC1
          C
 211            CALL SEGMT
 212            CALL SENGER
 213            CALL SEVOS
 214            CALL SESEA
 215            CALL FREQIN
 216            CALL ANLGIN
          CCCC  CALL FREQIN
          C
          C     ALLOW FOR PARK AND NUETRAL TRANMISSON SETTINGS
          C
 217            IF(DTRST.EQ.NTRAL) THEN
 218              DLCVV=0
 219              DHCVV=0
 220            ENDIF
          C
 221            IF((PRMOOP.NE.LAND).AND.(DTRST.NE.DRIVE)) DTDGS=LOW
 222            IF((PRMOOP.EQ.LAND).AND.(PDMOOP.EQ.TRNSTN)) DTDGS=LOW
 223            IF((DTRST.NE.NTRAL).AND.(DHCVV.EQ.0).AND.
               +(DLCVV.EQ.0)) NCLTCH=.TRUE.
   4            IF((PRMOOP.EQ.SEA).OR.(PDMOOP.EQ.SEA)) NCLTCH=.FALSE.
          C
          C     SHOULD WE CALL SHIFT
          C
 225            IF((PRMOOP.EQ.LAND).OR.(NCLTCH)) THEN
 226                IF(SFTINP) THEN
 227                    CALL SHIFT
 228                    GOTO 300
 229                ENDIF
 230                IF((DTDGS.NE.PTDGS).OR.(NCLTCH)) THEN
 231                    SFTINP=.TRUE.
 232                    PDTDGS=DTDGS
 233                    CALL SHIFT
 234                    IF(SFTINP)GOTO 300
 235                E.JDIF
 236            ENDIF
          C
          C     DO WE HAVE AN ACTIVE TIMER DELAY
          C
 237            IF(TDELAY.NE.0) THEN
 238                TIMER=TIMER+1
 239                IF(TIMER.GE.TDELAY) CALL CHOMP
 240                GOTO 200
 241            ENDIF
          C
          C     ARE WE PRESENTLY INVO' VED IN A MODE CHANGE ?
          C
 .42            IF(.NOT.MOOPCC) THEN
 243                CALL C.IOMP
 244                GOTO 200
```

```
             ENDIF
      C
      C      IS A MODE CHANGE DESIRED ?
      C
             IF(PRMODP.NE.DMODP) THEN
                PDMODP=DMODP
                MODPCC=.FALSE.
                CALL CHOMP
             ENDIF
      C
      200    CONTINUE
      C
      C      DID CHOMP CALL SHIFT ?
      C
             IF(SFTINP) GOTO 300
      C
             IF((DTRST.EQ.DRIVE).OR.(DTRST.EQ.REVRSE)) CALL DEMOS
      C
      C      IN DRIVE OR REVERSE CALL MOTREP
      C
             IF((DTRST.EQ.DRIVE).OR.(DTRST.EQ.REVRSE)) CALL MOTREP
      C
             CALL CCAN
      C
      C      IN DRIVE OR REVERSE CALL ANTSTL
      C

             IF((DTRST.EQ.DRIVE).OR.(DTRST.EQ.REVRSE)) CALL ANTSTL
      C
      300    CONTINUE
      C
             CALL PWMOUT
             CALL ERROUT
      C      CALL MLTOPE
             CALL SYNC2
      C
      C      TEST TO SEE IF ENGINE IS RUNNING
      C
             ENGNON=SBITS(DIGIN(JCARD3),WIDTH3,BEGIN3)
             LPCPSF=TBIT(ENGNON,BIT5)
             LSCPSF=TBIT(ENGNON,BIT6)
             IF(AENSP.LT.400.0)THEN
                DIGOUT(8)=POWER
                DO 400 I=4,7
                DIGOUT(I)=0
      400       CONTINUE
                I=0
                IDES=0
                PFTVV=0
                GFTVV=0
                SATVV=0
                PATVV=0
                GOTO 10
             ELSE
                DIGOUT(8)=NORMAL
                DO 500 I=4,7
                DIGOUT(I)=0
      500       CONTINUE
```

```
 281             I=0
   2             GOTO 100
   3          ENDIF
 284    600   CONTINUE
 285          END
```

```
001A  FA                      CLI
   B  2E8E161600              MOV     SS,CS:@@STACK$FRAME
   0  BC1000                  MOV     SP,@@STACK$OFFSET
0023  8BEC                    MOV     BP,SP
0025  2E8E1E1000              MOV     DS,CS:@@DATA$FRAME
002A  53                      PUSH    BX
002B  53                      PUSH    BX
002C  9A00000000              CALL    INITFP
0031  9A00000000              CALL    TQ_001
0036  FB                      STI
                                                ; STATEMENT # 151
0037  9A00000000              CALL    INIT
                                                ; STATEMENT # 152
003C  8D1E0000                LEA     BX,IAPBCN
0040  2E8E060000              MOV     ES,CS:@CONST
0045  06                      PUSH    ES        ; 1
0046  53                      PUSH    BX        ; 2
0047  9A00000000              CALL    START
                  ?10:
                                                ; STATEMENT # 154
004C  2E8E060200              MOV     ES,CS:@CONST+2H
0051  26C70660020100          MOV     ES:I,1H
0058  26C70662020000          MOV     ES:I+2H,0H
                  @@000000:
005F  2E8E060200              MOV     ES,CS:@CONST+2H
0064  268B066002              MOV     AX,ES:I
0069  268B166202              MOV     DX,ES:I+2H
   E  BB0300                  MOV     BX,3H
   1  B90000                  MOV     CX,0H
0074  50                      PUSH    AX        ; 1
0075  52                      PUSH    DX        ; 2
0076  9A00000000              CALL    TQ_150
007B  58                      POP     AX        ; 2
007C  5A                      POP     DX        ; 1
007D  7E03                    JLE     $+5H
007F  E92D00                  JMP     @@000001
0082  D1E2                    SAL     DX,1
0084  D1E2                    SAL     DX,1
0086  87DA                    XCHG    BX,DX
0088  2E8E060000              MOV     ES,CS:@CONST
008D  26C7479C0000            MOV     ES:DIGOUT[BX-4H],0H
0093  26C7479E0000            MOV     ES:DIGOUT[BX-2H],0H
                  ?20:
                                                ; STATEMENT # 156
0099  2E8E060200              MOV     ES,CS:@CONST+2H
009E  26810660020100          ADD     ES:I,1H
00A5  26811662020000          ADC     ES:I+2H,0H
00AC  E9B0FF                  JMP     @@000000
                  @@000001:
                                                ; STATEMENT # 157
00AF  2E8E060200              MOV     ES,CS:@CONST+2H
00B4  26C70660020000          MOV     ES:I,0H
00BB  26C70662020000          MOV     ES:I+2H,0H
                  @@000002:
   2  2E8E060200              MOV     ES,CS:@CONST+2H
00C7  268B066002              MOV     AX,ES:I
00CC  268B166202              MOV     DX,ES:I+2H
```

```
00D1   BB0F00             MOV     BX,0FH
   4   B90000             MOV     CX,0H
   7   50                 PUSH    AX         ; 1
00D8   52                 PUSH    DX         ; 2
00D9   9A00000000         CALL    TQ_150
00DE   58                 POP     AX         ; 2
00DF   5A                 POP     DX         ; 1
00E0   7E03               JLE     $+5H
00E2   E93A00             JMP     @@000003
00E5   87DA               XCHG    BX,DX
00E7   87C8               XCHG    CX,AX
00E9   B80200             MOV     AX,2H
00EC   BA0000             MOV     DX,0H
00EF   53                 PUSH    BX         ; 1
00F0   51                 PUSH    CX         ; 2
00F1   9A00000000         CALL    TQ_140
00F6   59                 POP     CX         ; 2
00F7   5B                 POP     BX         ; 1
00F8   D1E3               SAL     BX,1
00FA   D1E3               SAL     BX,1
00FC   2E8E060200         MOV     ES,CS:@CONST+2H
0101   26894704           MOV     ES:MASK[BX],AX
0105   26895706           MOV     ES:MASK[BX+2H],DX
                ?30:
                                             ; STATEMENT # 159
0109   2E8E060200         MOV     ES,CS:@CONST+2H
010E   26810660020100     ADD     ES:I,1H
   5   26811662020000     ADC     ES:I+2H,0H
   C   E9A3FF             JMP     @@000002
                @@000003:
                                             ; STATEMENT # 160
011F   2E8E060200         MOV     ES,CS:@CONST+2H
0124   26C70660020000     MOV     ES:I,0H
0128   26C70662020000     MOV     ES:I+2H,0H
0132   9B2ED9060A00       FLD     CS:@CONST+0AH; 7
0138   2E8E060600         MOV     ES,CS:@CONST+6H
013D   9B26D91E0000       FSTP    ES:M1      ; 7
0143   9B                 WAIT
0144   2E8E060800         MOV     ES,CS:@CONST+8H
0149   26C706A0000100     MOV     ES:INTIAL,1H
0150   26C706A2000000     MOV     ES:INTIAL+2H,0H
0157   2E8E060000         MOV     ES,CS:@CONST
015C   26A1069800         MOV     AX,ES:HYBPSS
0161   268B169A00         MOV     DX,ES:HYBPSS+2H
0166   2689064800         MOV     ES:DIGOUT+8H,AX
016B   2689164A00         MOV     ES:DIGOUT+0AH,DX
0170   9A00000000         CALL    ERROUT
                                             ; STATEMENT # 165
0175   9A00000000         CALL    SYNC1
                                             ; STATEMENT # 166
017A   9A00000000         CALL    SYNC2
                ?40:
                                             ; STATEMENT # 168
   F   9A00000000         CALL    SYNC1
                                             ; STATEMENT # 169
0184   2E8E060200         MOV     ES,CS:@CONST+2H
0189   268B06A001         MOV     AX,ES:JCARD3
```

```
018E  268B16A201       MOV    DX,ES:JCARD3+2H
0193  D1E0             SAL    AX,1
0195  D1E0             SAL    AX,1
0197  87D8             XCHG   BX,AX
0199  8D5F30           LEA    BX,DIGIN[BX-4H]
019C  2E8E060000       MOV    ES,CS:@CONST
01A1  06               PUSH   ES         ; 1
01A2  53               PUSH   BX         ; 2
01A3  8D1E5001         LEA    BX,WIDTH3
01A7  2E8E060200       MOV    ES,CS:@CONST+2H
01AC  06               PUSH   ES         ; 3
01AD  53               PUSH   BX         ; 4
01AE  8D1E1801         LEA    BX,BEGIN3
01B2  06               PUSH   ES         ; 5
01B3  53               PUSH   BX         ; 6
01B4  9A00000000       CALL   SBITS
01B9  9A00000000       CALL   MQERINT
01BE  2E8E060000       MOV    ES,CS:@CONST
01C3  268906A000       MOV    ES:ENGNON,AX
01C8  268916A200       MOV    ES:ENGNON+2H,DX
                                         ; STATEMENT # 170
01CD  8D1EA000         LEA    BX,ENGNON
01D1  2E8E060000       MOV    ES,CS:@CONST
01D6  06               PUSH   ES         ; 1
01D7  53               PUSH   BX         ; 2
01D8  8D1E5C00         LEA    BX,BIT5
01DC  2E8E060200       MOV    ES,CS:@CONST+2H
01E1  06               PUSH   ES         ; 3
01E2  53               PUSH   BX         ; 4
01E3  9A00000000       CALL   TBIT
01E8  2E8E060800       MOV    ES,CS:@CONST+8H
01ED  2689062800       MOV    ES:LPCPSF,AX
01F2  2689162A00       MOV    ES:LPCPSF+2H,DX
                                         ; STATEMENT # 171
01F7  8D1EA000         LEA    BX,ENGNON
01FB  2E8E060000       MOV    ES,CS:@CONST
0200  06               PUSH   ES         ; 1
0201  53               PUSH   BX         ; 2
0202  8D1E6000         LEA    BX,BIT6
0206  2E8E060200       MOV    ES,CS:@CONST+2H
020B  06               PUSH   ES         ; 3
020C  53               PUSH   BX         ; 4
020D  9A00000000       CALL   TBIT
0212  2E8E060800       MOV    ES,CS:@CONST+8H
0217  2689062C00       MOV    ES:LSCPSF,AX
021C  2689162E00       MOV    ES:LSCPSF+2H,DX
                                         ; STATEMENT # 172
0221  2E8E060000       MOV    ES,CS:@CONST
0226  268B06A400       MOV    AX,ES:ENGMSG
022B  268B16A600       MOV    DX,ES:ENGMSG+2H
0230  2689061800       MOV    ES:DIGOUT+18H,AX
0235  2689161A00       MOV    ES:DIGOUT+1AH,DX
023A  268B0EC400       MOV    CX,ES:IAENSP
023F  268B1EC600       MOV    BX,ES:IAENSP+2H
0244  87C1             XCHG   AX,CX
0246  87D3             XCHG   DX,BX
0248  BB5802           MOV    BX,258H
```

```
0248  B90C00              MOV     CX,0H
024E  9A00000000          CALL    TQ_100
 253  8956FE              MOV     [BP].@TEMP3+2H,DX
0256  8946FC              MOV     [BP].@TEMP3+4H,AX
0259  9BDD46FC            FLD     [BP].@TEMP3+4H
025D  9B2ED8360E00        FDIV    CS:@CONST+0EH; 7
0263  2E8E060400          MOV     ES,CS:@CONST+4H
0268  9B26D91E7400        FSTP    ES:AENSP; 7
026E  9B                  WAIT
026F  9B2ED9061200        FLD     CS:@CONST+12H; 7
0275  9B26D81E7400        FCOMP   ES:AENSP; 7
027B  50                  PUSH    AX         ; 1
027C  9BDDBEFAFF          FSTSW   [BP].@STACK+2H
0281  58                  POP     AX         ; 1
0282  9E                  SAHF
0283  7703                JA      $+5H
0285  E90300              JMP     @@000004
0288  9A00000000          CALL    SYNC2
                                                    ; STATEMENT # 176
028D  E9EFFE              JMP     ?40
                 @@000004:
                                                    ; STATEMENT # 178
0290  2E8E060000          MOV     ES,CS:@CONST
0295  268B068800          MOV     AX,ES:LOW
029A  268B168A00          MOV     DX,ES:LOW+2H
029F  268906F000          MOV     ES:PDTDGS,AX
02A4  268916F200          MOV     ES:PDTDGS+2H,DX
02A9  268B0E6800          MOV     CX,ES:LAND
 2AE  268B1E6A00          MOV     BX,ES:LAND+2H
02B3  26890EE400          MOV     ES:PRMOOP,CX
02B8  26891EE600          MOV     ES:PRMOOP+2H,BX
02BD  268B366C00          MOV     SI,ES:TRNSTN
02C2  268B3E6E00          MOV     DI,ES:TRNSTN+2H
02C7  2689360001          MOV     ES:DMOOP,SI
02CC  26893E0201          MOV     ES:DMOOP+2H,DI
02D1  268B069C00          MOV     AX,ES:HYNBPS
02D6  268B169E00          MOV     DX,ES:HYNBPS+2H
02DB  2689064800          MOV     ES:DIGOUT+8H,AX
02E0  2689164A00          MOV     ES:DIGOUT+0AH,DX
02E5  268B066000          MOV     AX,ES:SUSMSG
02EA  268B166200          MOV     DX,ES:SUSMSG+2H
02EF  2689065800          MOV     ES:DIGOUT+18H,AX
02F4  2689165A00          MOV     ES:DIGOUT+1AH,DX
02F9  268B062C01          MOV     AX,ES:SLOWER
02FE  268B162E01          MOV     DX,ES:SLOWER+2H
0303  2689064000          MOV     ES:DIGOUT,AX
0308  2689164200          MOV     ES:DIGOUT+2H,DX
030D  C70600000000        MOV     ITIMER,0H
                 ?50:
                                                    ; STATEMENT # 185
0313  9A00000000          CALL    SYNC1
                                                    ; STATEMENT # 186
0318  813E00002C01        CMP     ITIMER,12CH
031E  7503                JNZ     $+5H
 320  E90300              JMP     @@000005
0323  FF060000            INC     ITIMER
0327  2E8E060000          MOV     ES,CS:@CONST
```

```
032C  268B06F800           MOV    AX,ES:TDELAY
0331  268B16FA00           MOV    DX,ES:TDELAY+2H
0336  9A00000000           CALL   TQ_152
033B  7503                 JNZ    $+5H
033D  E93F00               JMP    @@000006
0340  2E8E060000           MOV    ES,CS:@CONST
0345  268B06FC00           MOV    AX,ES:TIMER
034A  268B16FE00           MOV    DX,ES:TIMER+2H
034F  81C00100             ADD    AX,1H
0353  81D20000             ADC    DX,0H
0357  268906FC00           MOV    ES:TIMER,AX
035C  268916FE00           MOV    ES:TIMER+2H,DX
0361  268B0EF800           MOV    CX,ES:TDELAY
0366  268B1EFA00           MOV    BX,ES:TDELAY+2H
036B  87D9                 XCHG   BX,CX
036D  9A00000000           CALL   TQ_150
0372  7D03                 JGE    $+5H
0374  E90500               JMP    @@000007
0377  9A00000000           CALL   CHOMP
                  @@000007:
                                                ; STATEMENT # 191
037C  E96E00               JMP    ?60
                  @@000006:
                                                ; STATEMENT # 193
037F  2E8E060800           MOV    ES,CS:@CONST+8H
0384  268B06EC00           MOV    AX,ES:MOOPCC
0389  268B16EE00           MOV    DX,ES:MOOPCC+2H
038E  D0D8                 RCR    AL,1
0390  7903                 JNS    $+5H
0392  E90300               JMP    @@000008
0395  9A00000000           CALL   CHOMP
                                                ; STATEMENT # 195
039A  E95000               JMP    ?60
                  @@000008:
                                                ; STATEMENT # 197
039D  2E8E060000           MOV    ES,CS:@CONST
03A2  268B06E400           MOV    AX,ES:PRMOOP
03A7  268B16E600           MOV    DX,ES:PRMOOP+2H
03AC  268B0E0001           MOV    CX,ES:DMOOP
03B1  268B1E0201           MOV    BX,ES:DMOOP+2H
03B6  87D9                 XCHG   BX,CX
03B8  53                   PUSH   BX          ; 1
03B9  51                   PUSH   CX          ; 2
03BA  9A00000000           CALL   TQ_150
03BF  58                   POP    AX          ; 2
03C0  5A                   POP    DX          ; 1
03C1  7503                 JNZ    $+5H
03C3  E92700               JMP    @@000009
03C6  2E8E060000           MOV    ES,CS:@CONST
03CB  2689160401           MOV    ES:PDMOOP,DX
03D0  2689060601           MOV    ES:PDMOOP+2H,AX
03D5  2E8E060800           MOV    ES,CS:@CONST+8H
03DA  26C706EC000000       MOV    ES:MOOPCC,0H
03E1  26C706EE000000       MOV    ES:MOOPCC+2H,0H
03E8  9A00000000           CALL   CHOMP
                  @@000009:
                  ?60:
```

```
                                                   ; STATEMENT # 203
 3ED  9A00000000          CALL    SYNC2
                                                   ; STATEMENT # 204
03F2  E91EFF              JMP     ?50
                @@000005:
                                                   ; STATEMENT # 206
03F5  2E8E060800          MOV     ES,CS:@CONST+8H
03FA  26C706A0000000      MOV     ES:INTIAL,0H
0401  26C706A2000000      MOV     ES:INTIAL+2H,0H
0408  2E8E060000          MOV     ES,CS:@CONST
040D  268B066400          MOV     AX,ES:NSUSMG
0412  268B166600          MOV     DX,ES:NSUSMG+2H
0417  2689065800          MOV     ES:DIGOUT+18H,AX
041C  2689165A00          MOV     ES:DIGOUT+1AH,DX
0421  26C706FC000000      MOV     ES:TIMER,0H
0428  26C706FE000000      MOV     ES:TIMER+2H,0H
                ?100:
                                                   ; STATEMENT # 210
042F  9A00000000          CALL    SYNC1
                                                   ; STATEMENT # 211
0434  9A00000000          CALL    SEGMT
                                                   ; STATEMENT # 212
0439  9A00000000          CALL    SENGER
                                                   ; STATEMENT # 213
043E  9A00000000          CALL    SEVOS
                                                   ; STATEMENT # 214
0443  9A00000000          CALL    SESEA
                                                   ; STATEMENT # 215
 448  9A00000000          CALL    FREQIN
                                                   ; STATEMENT # 216
044D  9A00000000          CALL    ANLGIN
                                                   ; STATEMENT # 217
0452  2E8E060000          MOV     ES,CS:@CONST
0457  268B06E000          MOV     AX,ES:DTRST
045C  268B16E200          MOV     DX,ES:DTRST+2H
 461  268B0E7800          MOV     CX,ES:NTRAL
0466  268B1E7A00          MOV     BX,ES:NTRAL+2H
046B  87D9                XCHG    BX,CX
046D  9A00000000          CALL    TQ_150
0472  7403                JZ      $+5H
0474  E92100              JMP     @@000010
0477  2E8E060000          MOV     ES,CS:@CONST
047C  26C70634010000      MOV     ES:DLCVV,0H
0483  26C70636010000      MOV     ES:DLCVV+2H,0H
048A  26C70638010000      MOV     ES:DHCVV,0H
0491  26C7063A010000      MOV     ES:DHCVV+2H,0H
                @@000010:
                                                   ; STATEMENT # 221
0498  2E8E060000          MOV     ES,CS:@CONST
049D  268B06E400          MOV     AX,ES:PRMOOP
04A2  268B16E600          MOV     DX,ES:PRMOOP+2H
04A7  268B0E6800          MOV     CX,ES:LAND
04AC  268B1E6A00          MOV     BX,ES:LAND+2H
 4B1  87D9                XCHG    BX,CX
 4B3  9A00000000          CALL    TQ_150
04B8  7503                JNZ     $+5H
04BA  E93E00              JMP     @@000011
```

```
04BD   2E8E060000          MOV     ES,CS:@CONST
04C2   268B06E000          MOV     AX,ES:DTRST
 4C7   268B16E200          MOV     DX,ES:DTRST+2H
04CC   268B0E8000          MOV     CX,ES:DRIVE
04D1   268B1E8200          MOV     BX,ES:DRIVE+2H
04D6   87D9                XCHG    BX,CX
04D8   9A00000000          CALL    TQ_150
04DD   7503                JNZ     $+5H
04DF   E91900              JMP     @@000011
04E2   2E8E060000          MOV     ES,CS:@CONST
04E7   268B068800          MOV     AX,ES:LOW
04EC   268B168A00          MOV     DX,ES:LOW+2H
04F1   268906EC00          MOV     ES:DTBOS,AX
04F6   268916EE00          MOV     ES:DTBOS+2H,DX
                   @@000011:
                                                        ; STATEMENT # 222
04FB   2E8E060000          MOV     ES,CS:@CONST
0500   268B06E400          MOV     AX,ES:PRMOOP
0505   268B16E600          MOV     DX,ES:PRMOOP+2H
050A   268B0E6800          MOV     CX,ES:LAND
050F   268B1E6A00          MOV     BX,ES:LAND+2H
0514   87D9                XCHG    BX,CX
0516   9A00000000          CALL    TQ_150
051B   7403                JZ      $+5H
051D   E93E00              JMP     @@000012
0520   2E8E060000          MOV     ES,CS:@CONST
0525   268B060401          MOV     AX,ES:PBMOOP
052A   268B160601          MOV     DX,ES:PBMOOP+2H
 52F   268B0E6C00          MOV     CX,ES:TRNSTN
0534   268B1E6E00          MOV     BX,ES:TRNSTN+2H
0539   87D9                XCHG    BX,CX
053B   9A00000000          CALL    TQ_150
0540   7403                JZ      $+5H
0542   E91900              JMP     @@000012
0545   2E8E060000          MOV     ES,CS:@CONST
054A   268B068800          MOV     AX,ES:LOW
054F   268B168A00          MOV     DX,ES:LOW+2H
0554   268906EC00          MOV     ES:DTBOS,AX
0559   268916EE00          MOV     ES:DTBOS+2H,DX
                   @@000012:
                                                        ; STATEMENT # 223
055E   2E8E060000          MOV     ES,CS:@CONST
0563   268B06E000          MOV     AX,ES:DTRST
0568   268B16E200          MOV     DX,ES:DTRST+2H
056D   268B0E7800          MOV     CX,ES:NTRAL
0572   268B1E7A00          MOV     BX,ES:NTRAL+2H
0577   87D9                XCHG    BX,CX
0579   9A00000000          CALL    TQ_150
057E   7503                JNZ     $+5H
0580   E94500              JMP     @@000013
0583   2E8E060000          MOV     ES,CS:@CONST
0588   268B063801          MOV     AX,ES:BHCVV
058D   268B163A01          MOV     DX,ES:BHCVV+2H
0592   9A00000000          CALL    TQ_152
 597   7403                JZ      $+5H
0599   E92C00              JMP     @@000013
059C   2E8E060000          MOV     ES,CS:@CONST
```

```
05A1   268B063401        MOV     AX,ES:DLCVV
05A6   268B163601        MOV     DX,ES:DLCVV+2H
05AE   9A00000000        CALL    TQ_152
05B0   7403              JZ      $+5H
05B2   E91300            JMP     @@000013
05B5   2E8E060800        MOV     ES,CS:@CONST+8H
05BA   26C7060C010100    MOV     ES:NCLTCH,1H
05C1   26C7060E010000    MOV     ES:NCLTCH+2H,0H
                  @@000013:
                                         ; STATEMENT # 224
05C8   2E8E060000        MOV     ES,CS:@CONST
05CD   268B06E400        MOV     AX,ES:PRMOOP
05D2   268B16E600        MOV     DX,ES:PRMOOP+2H
05D7   268B0E7000        MOV     CX,ES:SEA
05DC   268B1E7200        MOV     BX,ES:SEA+2H
05E1   87D9              XCHG    BX,CX
05E3   53                PUSH    BX          ; 1
05E4   51                PUSH    CX          ; 2
05E5   9A00000000        CALL    TQ_150
05EA   58                POP     AX          ; 2
05EB   5A                POP     DX          ; 1
05EC   7503              JNZ     $+5H
05EE   E92000            JMP     @@000015
05F1   2E8E060000        MOV     ES,CS:@CONST
05F6   268B0E0401        MOV     CX,ES:PDMOOP
05FB   268B1E0601        MOV     BX,ES:PDMOOP+2H
0600   87C1              XCHG    AX,CX
0602   87D3              XCHG    DX,BX
0604   9A00000000        CALL    TQ_150
0609   7503              JNZ     $+5H
060B   E90300            JMP     @@000015
060E   E91300            JMP     @@000014
                  @@000015:
0611   2E8E060800        MOV     ES,CS:@CONST+8H
0616   26C7060C010000    MOV     ES:NCLTCH,0H
061D   26C7060E010000    MOV     ES:NCLTCH+2H,0H
                  @@000014:
                                         ; STATEMENT # 225
0624   2E8E060000        MOV     ES,CS:@CONST
0629   268B06E400        MOV     AX,ES:PRMOOP
062E   268B16E600        MOV     DX,ES:PRMOOP+2H
0633   268B0E6800        MOV     CX,ES:LAND
0638   268B1E6A00        MOV     BX,ES:LAND+2H
063D   87D9              XCHG    BX,CX
063F   9A00000000        CALL    TQ_150
0644   7503              JNZ     $+5H
0646   E91900            JMP     @@000017
0649   2E8E060800        MOV     ES,CS:@CONST+8H
064E   268B060C01        MOV     AX,ES:NCLTCH
0653   268B160E01        MOV     DX,ES:NCLTCH+2H
0658   D0D8              RCR     AL,1
065A   7303              JNB     $+5H
065C   E90300            JMP     @@000017
065F   E99B00            JMP     @@000016
                  @@000017:
                                         ; STATEMENT # 226
0662   2E8E060800        MOV     ES,CS:@CONST+8H
```

```
0667   26F606A40001       TEST    ES:SFTINP,1H
066D   7503               JNZ     $+5H
066F   E90300             JMP     @@000018
0672   9A00000000         CALL    SHIFT
                                                 ; STATEMENT # 229
0677   E95602             JMP     ?300
                 @@000018:
                                                 ; STATEMENT # 230
067A   2E8E060000         MOV     ES,CS:@CONST
067F   268B06EC00         MOV     AX,ES:DTDOS
0684   268B16EE00         MOV     DX,ES:DTDOS+2H
0689   268B0EF400         MOV     CX,ES:PTDOS
068E   268B1EF600         MOV     BX,ES:PTDOS+2H
0693   87D9               XCHG    BX,CX
0695   9A00000000         CALL    TQ_150
069A   7403               JZ      $+5H
069C   E91900             JMP     @@000020
069F   2E8E060800         MOV     ES,CS:@CONST+8H
06A4   268B060C01         MOV     AX,ES:NCLTCH
06A9   268B160E01         MOV     DX,ES:NCLTCH+2H
06AE   D0D8               RCR     AL,1
06B0   7303               JNB     $+5H
06B2   E90300             JMP     @@000020
06B5   E94700             JMP     @@000019
                 @@000020:
                                                 ; STATEMENT # 231
06B8   2E8E060800         MOV     ES,CS:@CONST+8H
06BD   26C706A4000100     MOV     ES:SFTINP,1H
06C4   26C706A6000000     MOV     ES:SFTINP+2H,0H
06CB   2E8E060000         MOV     ES,CS:@CONST
06D0   268B06EC00         MOV     AX,ES:DTDOS
06D5   268B16EE00         MOV     DX,ES:DTDOS+2H
06DA   268906F000         MOV     ES:PDTDOS,AX
06DF   268916F200         MOV     ES:PDTDOS+2H,DX
06E4   9A00000000         CALL    SHIFT
                                                 ; STATEMENT # 234
06E9   2E8E060800         MOV     ES,CS:@CONST+8H
06EE   268B06A400         MOV     AX,ES:SFTINP
06F3   268B16A600         MOV     DX,ES:SFTINP+2H
06F8   D0D8               RCR     AL,1
06FA   7303               JNB     $+5H
06FC   E9D101             JMP     ?300
                 @@000019:
                 @@000018:
                                                 ; STATEMENT # 237
06FF   2E8E060000         MOV     ES,CS:@CONST
0704   268B06F800         MOV     AX,ES:TDELAY
0709   268B16FA00         MOV     DX,ES:TDELAY+2H
070E   9A00000000         CALL    TQ_152
0713   7503               JNZ     $+5H
0715   E93F00             JMP     @@000021
0718   2E8E060000         MOV     ES,CS:@CONST
071D   268B06FC00         MOV     AX,ES:TIMER
0722   268B16FE00         MOV     DX,ES:TIMER+2H
0727   81C00100           ADD     AX,1H
072B   81D20000           ADC     DX,0H
072F   268906FC00         MOV     ES:TIMER,AX
```

```
0734   268916FE00          MOV     ES:TIMER+2H,DX
0739   268B0EF800          MOV     CX,ES:TDELAY
073E   268B1EFA00          MOV     BX,ES:TDELAY+2H
0743   87D9                XCHG    BX,CX
0745   9A00000000          CALL    TQ_150
074A   7D03                JGE     $+5H
074C   E90500              JMP     @@000022
074F   9A00000000          CALL    CHOMP
                 @@000022:
                                                  ; STATEMENT # 240
0754   E96E00              JMP     ?200
                 @@000021:
                                                  ; STATEMENT # 242
0757   2E8E060800          MOV     ES,CS:@CONST+8H
075C   268B06EC00          MOV     AX,ES:MOOPCC
0761   268B16EE00          MOV     DX,ES:MOOPCC+2H
0766   D0D8                RCR     AL,1
0768   7303                JNB     $+5H
076A   E90300              JMP     @@000023
076D   9A00000000          CALL    CHOMP
                                                  ; STATEMENT # 244
0772   E95000              JMP     ?200
                 @@000023:
                                                  ; STATEMENT # 246
0775   2E8E060000          MOV     ES,CS:@CONST
077A   268B06E400          MOV     AX,ES:PRMOOP
077F   268B16E600          MOV     DX,ES:PRMOOP+2H
0784   268B0E0001          MOV     CX,ES:DMOOP
0789   268B1E0201          MOV     BX,ES:DMOOP+2H
078E   87D9                XCHG    BX,CX
0790   53                  PUSH    BX         ; 1
0791   51                  PUSH    CX         ; 2
0792   9A00000000          CALL    TQ_150
0797   58                  POP     AX         ; 2
0798   5A                  POP     DX         ; 1
0799   7503                JNZ     $+5H
079B   E92700              JMP     @@000024
079E   2E8E060000          MOV     ES,CS:@CONST
07A3   2689160401          MOV     ES:PDMOOP,DX
07A8   2689060601          MOV     ES:PDMOOP+2H,AX
07AD   2E8E060800          MOV     ES,CS:@CONST+8H
07B2   26C706EC000000      MOV     ES:MOOPCC,0H
07B9   26C706EE000000      MOV     ES:MOOPCC+2H,0H
07C0   9A00000000          CALL    CHOMP
                 @@000024:
                 ?200:
                                                  ; STATEMENT # 252
07C5   2E8E060800          MOV     ES,CS:@CONST+8H
07CA   268B06A400          MOV     AX,ES:SFTINP
07CF   268B16A600          MOV     DX,ES:SFTINP+2H
07D4   D0D8                RCR     AL,1
07D6   7303                JNB     $+5H
07D8   E9F500              JMP     ?300
07DB   2E8E060000          MOV     ES,CS:@CONST
07E0   268B0EE000          MOV     CX,ES:DTRST
07E5   268B1EE200          MOV     BX,ES:DTRST+2H
07EA   268B369000          MOV     SI,ES:DRIVE
```

```
07EF  268B3E8200         MOV     DI,ES:DRIVE+2H
07F4  87C1               XCHG    AX,CX
07F6  87D3               XCHG    DX,BX
07F8  87DE               XCHG    BX,SI
07FA  87CF               XCHG    CX,DI
07FC  50                 PUSH    AX         ; 1
07FD  52                 PUSH    DX         ; 2
07FE  9A00000000         CALL    TQ_150
0803  58                 POP     AX         ; 2
0804  5A                 POP     DX         ; 1
0805  7503               JNZ     $+5H
0807  E92000             JMP     @@000026
080A  2E8E060000         MOV     ES,CS:@CONST
080F  268B0E7C00         MOV     CX,ES:REVRSE
0814  268B1E7E00         MOV     BX,ES:REVRSE+2H
0819  87C2               XCHG    AX,DX
081B  87D9               XCHG    BX,CX
081D  9A00000000         CALL    TQ_150
0822  7503               JNZ     $+5H
0824  E90300             JMP     @@000026
0827  E90500             JMP     @@000025
                   @@000026:
                                            ; STATEMENT # 253
082A  9A00000000         CALL    DEMOS
                   @@000025:
                                            ; STATEMENT # 254
082F  2E8E060000         MOV     ES,CS:@CONST
0834  268B06E000         MOV     AX,ES:DTRST
0839  268B16E200         MOV     DX,ES:DTRST+2H
083E  268B0E8000         MOV     CX,ES:DRIVE
0843  268B1E8200         MOV     BX,ES:DRIVE+2H
0848  87D9               XCHG    BX,CX
084A  50                 PUSH    AX         ; 1
084B  52                 PUSH    DX         ; 2
084C  9A00000000         CALL    TQ_150
0851  58                 POP     AX         ; 2
0852  5A                 POP     DX         ; 1
0853  7503               JNZ     $+5H
0855  E92000             JMP     @@000028
0858  2E8E060000         MOV     ES,CS:@CONST
085D  268B0E7C00         MOV     CX,ES:REVRSE
0862  268B1E7E00         MOV     BX,ES:REVRSE+2H
0867  87C2               XCHG    AX,DX
0869  87D9               XCHG    BX,CX
086B  9A00000000         CALL    TQ_150
0870  7503               JNZ     $+5H
0872  E90300             JMP     @@000028
0875  E90500             JMP     @@000027
                   @@000028:
0878  9A00000000         CALL    MOTREF
                   @@000027:
                                            ; STATEMENT # 255
087D  9A00000000         CALL    ECAN
                                            ; STATEMENT # 256
0882  2E8E060000         MOV     ES,CS:@CONST
0887  268B06E000         MOV     AX,ES:DTRST
088C  268B16E200         MOV     DX,ES:DTRST+2H
```

```
0891  268B0E80 0         MOV      CX,ES:DRIVE
 896  268B1E8200         MOV      BX,ES:DRIVE+2H
 89B  87D9               XCHG     BX,CX
089D  50                 PUSH     AX          ; 1
089E  52                 PUSH     DX          ; 2
089F  9A00000000         CALL     TQ_150
08A4  58                 POP      AX          ; 2
08A5  5A                 POP      DX          ; 1
08A6  7503               JNZ      $+5H
08A8  E92000             JMP      @@000030
08AB  2E8E060000         MOV      ES,CS:@CONST
08B0  268B0E7C00         MOV      CX,ES:REVRSE
08B5  268B1E7E00         MOV      BX,ES:REVRSE+2H
08BA  87C2               XCHG     AX,DX
08BC  87D9               XCHG     BX,CX
08BE  9A00000000         CALL     TQ_150
08C3  7503               JNZ      $+5H
08C5  E90300             JMP      @@000030
08C8  E90500             JMP      @@000029
                 @@000030:
08CB  9A00000000         CALL     ANTSTL
                 @@000029:
                 ?300:
                                              ; STATEMENT # 258
08D0  9A00000000         CALL     PWMOUT
                                              ; STATEMENT # 259
08D5  9A00000000         CALL     ERROUT
                                              ; STATEMENT # 260
 8DA  9A00000000         CALL     SYNC2
                                              ; STATEMENT # 261
08DF  2E8E060200         MOV      ES,CS:@CONST+2H
08E4  268B06A001         MOV      AX,ES:JCARD3
08E9  268B16A201         MOV      DX,ES:JCARD3+2H
08EE  D1E0               SAL      AX,1
08F0  D1E0               SAL      AX,1
08F2  87D8               XCHG     DX,AX
08F4  8D5F30             LEA      BX,DIGIN[BX-4H]
08F7  2E8E060000         MOV      ES,CS:@CONST
08FC  06                 PUSH     ES          ; 1
08FD  53                 PUSH     BX          ; 2
08FE  8D1E5001           LEA      BX,WIDTH3
0902  2E8E060200         MOV      ES,CS:@CONST+2H
0907  06                 PUSH     ES          ; 3
0908  53                 PUSH     BX          ; 4
0909  8D1E1801           LEA      BX,BEGIN3
090D  06                 PUSH     ES          ; 5
090E  53                 PUSH     BX          ; 6
090F  9A00000000         CALL     SBITS
0914  9A00000000         CALL     MQERINT
0919  2E8E060000         MOV      ES,CS:@CONST
091E  268906A000         MOV      ES:ENONON,AX
0923  268916A200         MOV      ES:ENONON+2H,DX
                                              ; STATEMENT # 262
 928  8D1EA000           LEA      BX,ENONON
 92C  2E8E060000         MOV      ES,CS:@CONST
0931  06                 PUSH     ES          ; 1
0932  53                 PUSH     BX          ; 2
```

```
0933  8D1E5C00          LEA     BX,BIT5
0937  2E8E060200        MOV     ES,CS:@CONST+2H
093C  06                PUSH    ES          ; 3
093D  53                PUSH    BX          ; 4
093E  9A00000000        CALL    TBIT
0943  2E8E060800        MOV     ES,CS:@CONST+8H
0948  2689062800        MOV     ES:LPCPSF,AX
094D  2689162A00        MOV     ES:LPCPSF+2H,DX
                                            ; STATEMENT # 263
0952  8D1EA000          LEA     BX,ENGNON
0956  2E8E060000        MOV     ES,CS:@CONST
095B  06                PUSH    ES          ; 1
095C  53                PUSH    BX          ; 2
095D  8D1E6000          LEA     BX,BIT6
0961  2E8E060200        MOV     ES,CS:@CONST+2H
0966  06                PUSH    ES          ; 3
0967  53                PUSH    BX          ; 4
0968  9A00000000        CALL    TBIT
096D  2E8E060800        MOV     ES,CS:@CONST+8H
0972  2689062C00        MOV     ES:LSCPSF,AX
0977  2689162E00        MOV     ES:LSCPS. +2H,DX
                                            ; STATEMENT # 264
097C  9B2ED9061200      FLD     CS:@CONST+12H; 7
0982  2E8E060400        MOV     ES,CS:@CONST+4H
0987  9B26D81E7400      FCOMP   ES:AENSP; 7
098D  50                PUSH    AX          ; 1
098E  9BDDBEFAFF        FSTSW   [BP].@STACK+2H
0993  58                POP     AX          ; 1
0994  9E                SAHF
0995  7703              JA      $+5H
0997  E9DD00            JMP     @@000031
099A  2E8E060200        MOV     ES,CS:@CONST+2H
099F  268B063C02        MOV     AX,ES:POWER
09A4  268B163A02        MOV     DX,ES:POWER+2H
09A9  2E8E060000        MOV     ES,CS:@CONST
09AE  2689065C00        MOV     ES:DIGOUT+1CH,AX
09B3  2689165E00        MOV     ES:DIGOUT+1EH,DX
09B8  2E8E060200        MOV     ES,CS:@CONST+2H
09BD  26C70660020400    MOV     ES:I,4H
09C4  26C70662020000    MOV     ES:I+2H,0H
                  @@000032:
                                            ; STATEMENT # 266
09CB  2E8E060200        MOV     ES,CS:@CONST+2H
09D0  268B066002        MOV     AX,ES:I
09D5  268B166202        MOV     DX,ES:I+2H
09DA  BB0700            MOV     BX,7H
09DD  B90000            MOV     CX,0H
09E0  50                PUSH    AX          ; 1
09E1  52                PUSH    DX          ; 2
09E2  9A00000000        CALL    TQ.150
09E7  58                POP     AX          ; 2
09E8  5A                POP     DX          ; 1
09E9  7E03              JLE     $+5H
09EB  E92D00            JMP     @@0000 33
09EE  D1E2              SAL     DX,1
09F0  D1E2              SAL     DX,1
09F2  87DA              XCHG    DX,BX
```

```
09F4   2E8E060000         MOV      ES,CS:@CONST
09F9   26C7473C0000       MOV      ES:DIGOUT[BX-4H],0H
09FF   26C7473E0000       MOV      ES:DIGOUT[BX-2H],0H
                  ?400:
                                                  ; STATEMENT # 268
0A05   2E8E060200         MOV      ES,CS:@CONST+2H
0A0A   26810660020100     ADD      ES:I,1H
0A11   26811662020000     ADC      ES:I+2H,0H
0A18   E9B0FF             JMP      @@000032
                  @@000033:
                                                  ; STATEMENT # 269
0A1B   B80000             MOV      AX,0H
0A1E   BA0000             MOV      DX,0H
0A21   2E8E060200         MOV      ES,CS:@CONST+2H
0A26   2689066002         MOV      ES:I,AX
0A2B   2689166202         MOV      ES:I+2H,DX
0A30   89C1               MOV      CX,AX
0A32   2E8E060000         MOV      ES,CS:@CONST
0A37   2689067C01         MOV      ES:IDES,AX
0A3C   26890E7E01         MOV      ES:IDES+2H,CX
0A41   89C3               MOV      BX,AX
0A43   2689065001         MOV      ES:PFTVV,AX
0A48   26891E5201         MOV      ES:PFTVV+2H,BX
0A4D   89C6               MOV      SI,AX
0A4F   2689065801         MOV      ES:SFTVV,AX
0A54   2689365A01         MOV      ES:SFTVV+2H,SI
0A59   89C7               MOV      DI,AX
0A5B   2689065401         MOV      ES:SATVV,AX
0A60   26893E5601         MOV      ES:SATVV+2H,DI
0A65   89C2               MOV      DX,AX
0A67   2689064C01         MOV      ES:PATVV,AX
0A6C   2689164E01         MOV      ES:PATVV+2H,DX
0A71   E9D3F5             JMP      ?10
                                                  ; STATEMENT # 276
0A74   E99700             JMP      @@000034
                  @@000031:
                                                  ; STATEMENT # 277
0A77   2E8E060200         MOV      ES,CS:@CONST+2H
0A7C   268B063402         MOV      AX,ES:NORMAL
0A81   268B163602         MOV      DX,ES:NORMAL+2H
0A86   2E8E060000         MOV      ES,CS:@CONST
0A8B   2689063C00         MOV      ES:DIGOUT+1CH,AX
0A90   2689163E00         MOV      ES:DIGOUT+1EH,DX
0A95   2E8E060200         MOV      ES,CS:@CONST+2H
0A9A   26C70660020400     MOV      ES:I,4H
0AA1   26C70662020000     MOV      ES:I+2H,0H
                  @@000035:
                                                  ; STATEMENT # 278
0AA8   2E8E060200         MOV      ES,CS:@CONST+2H
0AAD   268B066002         MOV      AX,ES:I
0AB2   268B166202         MOV      DX,ES:I+2H
0AB7   BB0700             MOV      BX,7H
0ABA   B90000             MOV      CX,0H
0ABD   50                 PUSH     AX          ; 1
0ABE   52                 PUSH     DX          ; 2
0ABF   9A00000000         CALL     TQ_I50
0AC4   58                 POP      AX          ; 2
```

```
0AC5   5A                         POP       DX              ; 1
0AC6   7E03                       JLE       $+5H
0AC8   E92D00                     JMP       @@000036
0ACB   D1E2                       SAL       DX,1
0ACD   D1E2                       SAL       DX,1
0ACF   87DA                       XCHG      BX,DX
0AD1   2E8E060000                 MOV       ES,CS:@CONST
0AD6   26C7479C0000               MOV       ES:BIGOUT[BX-4H],0H
0ADC   26C7479E0000               MOV       ES:BIGOUT[BX-2H],0H
                    ?500:
                                                         ; STATEMENT # 280
0AE2   2E8E060200                 MOV       ES,CS:@CONST+2H
0AE7   26810660020100             ADD       ES:I,1H
0AEE   26811662020000             ADC       ES:I+2H,0H
0AF5   E9D0FF                     JMP       @@000035
                    @@000036:
                                                         ; STATEMENT # 281
0AF8   2E8E060200                 MOV       ES,CS:@CONST+2H
0AFD   26C70660020000             MOV       ES:I,0H
0B04   26C70662020000             MOV       ES:I+2H,0H
0B0B   E921F9                     JMP       ?100
                    @@000034:
                    ?600:
                                                         ; STATEMENT # 285
0B0E   9A00000000                 CALL      TQ_999
```

```
FORTRAN-86 COMPILER       ERROR MESSAGES AND SUMMARY
:F2:ATR.FOR

STORAGE REQUIREMENTS FOR MODULE ATR:

          CODE AREA SIZE          00AFDH      2813D
          CONSTANT AREA SIZE      00016H        22D
          VARIABLE AREA SIZE      00002H         2D
          MAXIMUM STACK SIZE      00010H        16D
          /ERROR/                 0012CH       300D
          /MOUT/                  000C6H       198D
          /CINOUT/                0019CH       412D
          /BITFNC/                00268H       616D
          /CALC/                  0020CH       524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 1AH
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF ATR COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY: :F1:FORT86.86  :F2:BITFNC.FOR

          LOGICAL*4 FUNCTION TBIT(NAME,BIT)
          INTEGER*4 RESULT,MASK(16),NAME,BIT
          COMMON /BITFNC/ DUMMY,MASK
          RESULT=NAME.AND.MASK(BIT)
          IF(RESULT.EQ.0) THEN
               TBIT=.FALSE.
               ELSE
               TBIT=.TRUE.
          ENDIF
          RETURN
          END
```

```
      FUNCTION SBITS(NAME,WIDTH,BEGIN)
      INTEGER*4 RESULT,MASK(16),NAME,WIDTH,BEGIN,RMASK,END
      COMMON /BITFNC/ RESULT,MASK
      RMASK=0
      END=BEGIN+(WIDTH-1)
      DO 10 K=BEGIN,END
      RMASK=RMASK.OR.MASK(K)
   10 CONTINUE
      SBITS=NAME.AND.RMASK
      RETURN
      END
```

```
  23           BLOCKDATA DATATR
    =1  $      INCLUDE (:F2:DATATR.FOR)
   4=1         INTEGER*4 RESULT,MASK(16),NAME
  25=1         INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
  26=1         INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
  27=1         INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
  28=1         INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
  29=1         INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
  30=1         INTEGER*4 BIT19,BIT29,BIT39,BIT49
  31=1         INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
  32=1         INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
  33=1         INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
  34=1         INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
  35=1         INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
  36=1         INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
  37=1         INTEGER*4 D01,D02,D03,D04,D05,D06,D07,D08,D09,D010
  38=1         INTEGER*4 D011,D012,D013,D014,D015,D016,D017,D018,D019,D020
  39=1         INTEGER*4 D021,D022,D023,D024,D025,NORMAL,POWER
  40=1         INTEGER*4 D026,D027,D028,D029,D030
  41=1         INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
  42=1         LOGICAL*4 MSSF,TSSF,OSSF,UKNPSF,P83CSF,INDPSF
  43=1         LOGICAL*4 HEOTSF,LBCLSF,LSKPSF,LSSPSF,LPCPSF
  44=1         LOGICAL*4 LSCPSF,HPBPSF,LEOPSF,LPHLSF,LSHLSF
  45=1         LOGICAL*4 TBIT,LSPPSF,FIRESP
  46=1         LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  47=1         LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  48=1         LOGICAL*4 LVSVSP,LCSVSP,HVSVSP,HCSVSP
   9=1         LOGICAL*4 AEBPSF,SEWPSF,HDPVSF,FEBPSF,PGCSSF
   5=1         LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
  51=1         LOGICAL*4 APBCSP,ASBCSP,DHMHSP,DHMLSP,DHMRSP,DTRNSP
  52=1         LOGICAL*4 PAMHSP,PAMLSP,PFMHSP,PFMLSP,SFMHSP,SFMLSP
  53=1         LOGICAL*4 SAMHSP,SAMLSP,ENCTSP,PHOTSP,SHOTSP
  54=1         LOGICAL*4 MOOPCC,MLTSSP,RENTRY
  55=1         LOGICAL*4 NOFAN,F40PM,F80PM,F120 M,FTIME,NCLTCH
  56=1         LOGICAL*4 AENSP,APSSP,APMSP,ASSSP,ASMSP,APWSP,ASWSP
  57=1         INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
  58=1         INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
  59=1         INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  60=1         INTEGER*4 TRNDIR
  61=1         INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
  62=1         INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  63=1         INTEGER*4 TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  64=1         INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
  65=1         INTEGER*4 CKPSLT,CKRSL1,CKRSL2
  66=1         INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  67=1         INTEGER*4 DTRST,PRMOOP
  68=1         INTEGER*4 TDELAY,TIMER,DMOOP,PDMOOP,DIGOUT(3)
  69=1         INTEGER*4 SECFOP,PCFWNP,DLGPON
  70=1         INTEGER*4 GCTD,GRCLSE,GOTD,GROPEN,PCFWDP,DLGOFF,SLOWER
  71=1         INTEGER*4 SUBDWN,DLEVV,DHEVV,MNBPON,MBPON,TRKSTP,SURISE
  72=1         INTEGER*4 ERRDLY,DTBSS,PDTBSS,PTBSS
  73=1         INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  74=1         INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
   5=1         INTEGER*4 PTVV,STVV
   5=1         REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  77=1         REAL*4 FDR,WJCNST,INTCPT(-1:1)
  78=1         REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  79=1         REAL*4 DPMSP,DSMSP,DDHMSP,PNT16,AUX9,AUX10
```

```
   80=1          REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
   81=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
   82=1          REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
   83=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
   84=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
   85=1          REAL*4 DPFRT,PDIFF,DPPWJ,DSFRT,SDIFP,DPSWJ
   86=1          REAL*4 PPDIS,PPDFP,APSSP,ASSSP
   87=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
   88=1          REAL*4 SPDIS,SPDFP,MAXMSP,TREF,DES
   89=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   90=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   91=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   92=1          REAL*4 DHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   93=1          COMMON /BITFNC/ RESULT,MASK,NAME
   94=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   95=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   96=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   97=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   98=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   99=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
  100=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
  101=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
  102=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
  103=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
  104=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
  105=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
  106=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
  107=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
  108=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
  109=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
  110=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
  111=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
  112=1          COMMON /ERROR/MSSF,TSSF,GSSF,DKNFSF,PGSCSF,INDFSF
  113=1          COMMON /ERROR/ HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
  114=1          COMMON /ERROR/ LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
  115=1          COMMON /ERROR/ LSPPSF,FIRESF
  116=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  117=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  118=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  119=1          COMMON /ERROR/ AEBPSF,SEWPSF,H3PVSF,FEDPSF,PGCSSF
  120=1          COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
  121=1          COMMON /ERROR/ APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
  122=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  123=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  124=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
  125=1          COMMON /ERROR/ NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
  126=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  127=1          COMMON /CINOUT/ IAPBCN,IASBCN,IDHMSF,IDHMSR,IPAMPR,IPFMPR
  128=1          COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  129=1          COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  130=1          COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  131=1          COMMON /CINOUT/ TEST1,TEST2,TEST3,HY3PSS,HYNDPS,ENGNON,ENGMSG
  132=1          COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  133=1          COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  134=1          COMMON /CINOUT/ IAENSF,IAPMSF,IASMSF,IAPSSF,IASSSF,IAPWSF,IASWSF
  135=1          COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
  136=1          COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
  137=1          COMMON /CINOUT/ SECPOP,PCPWNP,DLOPON
```

```
   138=1        COMMON /CINOUT/ GCTD,GRCLSE,GOTD,GROPEN,FCFWDF,BLGOFF,SLOWER
   139=1        COMMON /CINOUT/ SUDOWN,DLCVV,DHCVV,MNSPON,MDPON,TRKSTP,SURISE
   140=1        COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
   141=1        COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDIS
   142=1        COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
   143=1        COMMON /CINOUT/ PTVV,STVV
   144=1        COMMON /CALC/ APBCAN,ASBCAN,BHMSP,BHMSR,BTRNR,PAMTPR
   145=1        COMMON /CALC/ FBR,WJCNST
   146=1        COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENGTEM,PHSTEM,SHSTEM
   147=1        COMMON /CALC/ BPMSP,BSMSP,BDHMSP,PNT1G,AUX9,AUX10
   148=1        COMMON /CALC/ BHWSP,BHWSR,BPWSP,BSWSP,INTCPT
   149=1        COMMON /CALC/ APPG,ASPG,AENSP,PMDIS,PMDFP
   150=1        COMMON /CALC/ PMTRQ,BPMP,REPPT,PTREFF,REPST
   151=1        COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
   152=1        COMMON /CALC/ SMTRQ,BSMP,STREFF,REPT
   153=1        COMMON /CALC/ BPFRT,PDIFF,BPPWJ,BSFRT,SDIFF,BPSWJ
   154=1        COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
   155=1        COMMON /CALC/ PPTRQ,BPPP,REPPP,PPMEFF,REPSP
   156=1        COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
   157=1        COMMON /CALC/ SPTRQ,BSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   158=1        COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   159=1        COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   160=1        COMMON /CALC/ BHMSPB,PAMPR,SAMPR,PFMPR,SFMPR
   161=1        DATA RESULT,MASK,NAME/18*0/
   162=1        DATA BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
      =1       +/0,0,1,2,3,4,5,14,0,8/
   163=1        DATA BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
      =1       +/7,8,0,7,8,9,10,11/
   164=1        DATA BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
      =1       +/3,4,5,10,11,12,13,1/
   165=1        DATA BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
      =1       +/4,14,15,6,15,0,0,0/
   166=1        DATA BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47/8*0/
   167=1        DATA BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
      =1       +/2,3,4,12,1,2,0,0/
   168=1        DATA BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
      =1       +/7,10,1,6,7,1,1/
   169=1        DATA BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
      =1       +/14,5,4*0,10/
   170=1        DATA WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
      =1       +/3,4,6,3,5,6,4/
   171=1        DATA WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
      =1       +/2,2,4*0,4/
   172=1        DATA BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
      =1       +/14,0,2,0,2,0/
   173=1        DATA JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
      =1       +/3,3,1,2,1,2,3/
   174=1        DATA JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
      =1       +/3,3,4*0,2/
   175=1        DATA BG1,BG2,BG3,BG4,BG5,BG6,BG7,BG8,BG9,BG10
      =1       +/18,172,17,164,9,18,168,17,2,74/
   176=1        DATA BG11,BG12,BG13,BG14,BG15,BG16,BG17,BG18,BG19,BG20
      =1       +/49,18,164,136,9,18,164,136,17,12/
   177=1        DATA BG21,BG22,BG23,BG24,BG25,NORMAL,POWER/4,8,12,249,4,1,2/
   178=1        DATA BG26,BG27,BG28,BG29,BG30/2,31,64,128,243/
   179=1        DATA WIDTH,BEGIN,END,RMASK,I,K/6*0/
   180=1        DATA MSSF,TSSF,GSSF,BKNFSF,POSCSF,INEPSF/6*.FALSE./
   181=1        DATA HEOTSF,LECLSF,LERPSF,LESPSF,LPCPSF/5*.FALSE./
```

```
      DATA LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF/5*.FALSE./
      DATA LSPPSF,FIRESF/2*.FALSE./
      DATA P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF/5*.FALSE./
      DATA S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF/5*.FALSE./
      DATA LVSVSF,LCSVSF,HVSVSF,HCSVSF/4*.FALSE./
      DATA AEBPSF,SEWPSF,HBPVSF,FEBPSF,POCSSF/5*.FALSE./
      DATA RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP/5*.FALSE./
      DATA APBLSF,ASBCSF,DAMASF,DAMLSF,DAMRSF,DTRNSF/6*.FALSE./
      DATA PAMASF,PAMLSF,PFMASF,PFMLSF,SFMASF,SFMLSF/6*.FALSE./
      DATA SAMASF,SAMLSF,ENCTSF,PAOTSF,SAOTSF/5*.FALSE./
      DATA MOOPCC,MLTSSF,RENTRY/3*.FALSE./
      DATA NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCA/6*.FALSE./
      DATA AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF/7*.FALSE./
      DATA IAPBCN,IASBCN,IDAMSP,IDAMSR,IPAMPR,IPFMPR/6*0/
      DATA ISFMPR,ISAMPR,IENCTM,IPAOTM,ISAOTM,IDTRNR/6*0/
      DATA TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
     +/0,11*0,32768,0/
      DATA LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
     +/256,128,64,4096,1024,2048,512,16,32/
      DATA TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG/3*0,96,0,0
     +,16384/
      DATA CKPUMP,CKRAMP,CKPLST,CKRLST/2*0,11,24576/
      DATA CKPSLT,CKRSLT,CKRSL2/0,0,16384/
      DATA IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP/7*0/
      DATA DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS/2*0,50,3*0/
      DATA TDELAY,TIMER,DMOOP,PDMOOP/4*0/
      DATA SECFOF,PCFWNP,BLGPON/3*0/
      DATA GCTD,GRCLSE,GOTD,GROPEN,PCFWBP,BLGOFF,SLOWER/50,0,30,3*0,2/
      DATA SODOWN,DLCVV,DHCVV,MN3PON,M3PON,TRKSTP,SORISE/300,4*0,50,0/
      DATA PATVV,PFTVV,SATVV,SFTVV/4*0/
      DATA AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES/9*0/
      DATA RSMS,RPMS,RSSS,RPSS,RES/5*0/
      DATA PTVV,STVV/2*0/
      DATA APBCAN,ASBCAN,DAMSP,DAMSR,DTRNR,PAMTPR/6*0.0/
      DATA FDR,WJCNST
     +/10.88,0.5676/
      DATA PFMTPR,SFMTPR,SAMTPR,ENCTEM,PAOTEM,SAOTEM/6*0.0/
      DATA DPMSP,DSMSP,ODAMSP,PNTIG,AUX9,AUX10/4*0.0,0.186,0.[illegible]17/
      DATA DAWSP,DAWSR,DPWSP,DSWSP,INTCPT/4*0.0,2.587,0.0,-0.4/
      DATA APPS,ASPS,AENSP,PMDIS,PMDFP/5*0.0/
      DATA PMTRQ,DPMP,REPPT,PTREFF,REPST/3*0.0,.85,0.0/
      DATA APMSP,ASMSP,SMDIS,SMDFP/4*0.0/
      DATA SMTRQ,DSMP,STREFF,REPT/2*0.0,.85,0.0/
      DATA DPFRT,PDIFP,DPFWJ,DSFRT,SDIFP,DPSWJ/6*0.0/
      DATA PPDIS,PPDFP,APSSP,ASSSP/4*0.0/
      DATA PPTRQ,DPPP,REPPP,PPMEFF,REPSP/3*0.0,.95,0.0/
      DATA SPDIS,SPDFP,MAXMSP,TREP,DES/5*0.0/
      DATA SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP/2*0.0,.95,3*0.0/
      DATA ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE/6*0.0/
      DATA K1,K2,K3,K4,K5,K6,K7,K8,K9,K10/0.18,-0.36,1.25,0.7575
     +,-0.70,0.439,1.02,3*0.0/
      DATA DAMSPB,PAMPR,SAMPR,PFMPR,SFMPR/44*0.0/
      END
```

```
FORTRAN-86 COMPILER        ERROR MESSAGES AND SUMMARY
:F2:BITFNC.FOR

STORAGE REQUIREMENTS FOR MODULE TBIT:

    CODE AREA SIZE         00074H       116D
    CONSTANT AREA SIZE     00002H         2D
    VARIABLE AREA SIZE     00003H         3D
    MAXIMUM STACK SIZE     00003H         3D
    /BITFNC/               00044H        68D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 4H
COMPILATION OF TBIT COMPLETE.
```

```
FORTRAN-86 COMPILER       ERROR MESSAGES AND SUMMARY
:F2:BITFNC.FOR

STORAGE REQUIREMENTS FOR MODULE SBITS:

     CODE AREA SIZE        000ACH       172D
     CONSTANT AREA SIZE    00002H         2D
     VARIABLE AREA SIZE    0000EH        14D
     MAXIMUM STACK SIZE    0000EH        14D
     /BITFNC/              00044H        68D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 4H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF SBITS COMPLETE.
```

```
STORAGE REQUIREMENTS FOR MODULE DATATR:

      CODE AREA SIZE         00000H          0D
      CONSTANT AREA SIZE     00000H          0D
      VARIABLE AREA SIZE     00004H          4D
      MAXIMUM STACK SIZE     00000H          0D
      /ERROR/                0012CH        300D
      /CINOUT/               0019CH        412D
      /BITFNC/               00268H        616D
      /CALC/                 0020CH        524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
COMPILATION OF DATATR COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
  PILER INVOKED BY: :F1:FORT86.86  :F2:DNSHFT.FOR

     1          SUBROUTINE DNSHFT
    =1 $        INCLUDE(:F2:COMMON.FOR)
    2=1         INTEGER*4 RESULT,MASK(16),NAME
    3=1         INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
    4=1         INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
    5=1         INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
    6=1         INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
    7=1         INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
    8=1         INTEGER*4 BIT19,BIT29,BIT39,BIT49
    9=1         INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
   10=1         INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
   11=1         INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
   12=1         INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
   13=1         INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
   14=1         INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
   15=1         INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   16=1         INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   17=1         INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   18=1         INTEGER*4 DG26,DG27,DG28,DG29,DG30
   19=1         INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
   20=1         LOGICAL*4 MSSF,TSSF,GSSF,BKNPSF,FGSCSF,INBPSF
   21=1         LOGICAL*4 HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
   22=1         LOGICAL*4 LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
   23=1         LOGICAL*4 TBIT,LSPPSF,FIRESF
   24=1         LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
   25=1         LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
   26=1         LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
   27=1         LOGICAL*4 AEBPSF,SEWPSF,H3PVSF,FEDPSF,PGCSSF
   28=1         LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
   29=1         LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
   30=1         LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
   31=1         LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
   32=1         LOGICAL*4 MOOPCC,MLTSSF,RENTRY
   33=1         LOGICAL*4 NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
   34=1         LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
   35=1         INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
   36=1         INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
   37=1         INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
   38=1         INTEGER*4 TRNDIR
   39=1         INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
   40=1         INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
   41=1         INTEGER*4 TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
   42=1         INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
   43=1         INTEGER*4 CKPSLT,CKRSL1,CKRSL2
   44=1         INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
   45=1         INTEGER*4 DTRST,PRMODP
   46=1         INTEGER*4 TDELAY,TIMER,DMODP,PDMODP,DIGOUT(3)
   47=1         INTEGER*4 SECFOF,PCFWNP,BLOPON
   48=1         INTEGER*4 GCTD,GRCLSE,GOTD,GROPEN,PCFWDP,BLGOFF,SLOWER
   49=1         INTEGER*4 SUDOWN,DLCVV,DHCVV,MNSPON,MSPON,TRNSTP,SURISE
   50=1         INTEGER*4 EPRDLY,DTDGS,PDTDGS,PTDGS
   51=1         INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
   52=1         INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
   53=1         INTEGER*4 PTVV,STVV
   54=1         REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
   55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
   56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1          REAL*4 BPMSP,BSMSP,BDHMSP,FNT10,AUX9,AUX10
   58=1          REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
   59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
   60=1          REAL*4 PMTRQ,BPMP,REPPT,PTREFF,REPST
   61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
   62=1          REAL*4 SMTRQ,BSMP,STREFF,REPT
   63=1          REAL*4 BPFRT,PDIFP,BPPWJ,BSFRT,SDIFP,BPSWJ
   64=1          REAL*4 PPDIS,PPDFP,APSSP,ASSSP
   65=1          REAL*4 PPTRQ,BPPP,REPPP,PPMEFF,REPSP
   66=1          REAL*4 SPDIS,SPDFP,MAXMSP,TREP,BES
   67=1          REAL*4 SPTRQ,BSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1          REAL*4 M29,M30,M31,M32,M33,M34
   74=1          REAL*4 BHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   75=1          INTEGER*4 M2,M3,M10,M11,M12
   76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1          COMMON /BITFNC/ RESULT,MASK,NAME
   79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1          COMMON /ERROR/MSSF,TSSF,OSSF,BKNPSF,PGSCSF,INDPSF
   98=1          COMMON /ERROR/ HEOTSF,LBCLSF,LSRPSF,LSSPSF,LPCPSF
   99=1          COMMON /ERROR/ LSCPSF,HPEPSF,LEOPSF,LPHLSF,LSHLSF
  100=1          COMMON /ERROR/ LSPPSF,FIRESF
  101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,POCSSF
  105=1          COMMON /ERROR/ RGLSSF,LGTRNS,SLTRNS,INTIAL,SFTINP
  106=1          COMMON /ERROR/ APBCSF,ASBCSF,BHMHSF,BHMLSF,BHMRSF,BTRNSF
  107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  109=1          COMMON /ERROR/ MOOPCC,MLTSSP,RENTRY
  110=1          COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,PTIME,NCLTCH
  111=1          COMMON /ERROR/  ENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  112=1          COMMON /CINOUT/ IAPBCN,IASBCN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
```

```
      COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
      COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SDSMSG,NSDSMG
      COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
      COMMON /CINOUT/ TEST1,TEST2,TEST3,HYBPSS,HYNDPS,ENGNON,ENGMSG
      COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
      COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
      COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
      COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
      COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
      COMMON /CINOUT/ SECFOF,PCFWNP,BLGPON
      COMMON /CINOUT/ GCTD,GRCLSE,GOTD,GROPEN,PCFWDP,BLGOFF,SLOWER
      COMMON /CINOUT/ SUDOWN,DLCVV,DMCVV,MNBPON,MBPON,TRKSTP,SURISE
      COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
      COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
      COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
      COMMON /CINOUT/ PTVV,STVV
      COMMON /CALC/ APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
      COMMON /CALC/ PDR,WJCNST
      COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
      COMMON /CALC/ DPMSP,DSMSP,DDHMSP,PNT1G,AUX9,AUX10
      COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
      COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
      COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
      COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
      COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
      COMMON /CALC/ DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
      COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
      COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
      COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
      COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
      COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
      COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
      COMMON /CALC/ DHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
      COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
      COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
      COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
      COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
      COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
C
C     TEST TO SEE IF THE SHIFT IS ALLOWABLE
C
      IF(.NOT.PTIME) THEN
         FFDR=10.48
         DPMSP=APSSP*FFDR
         DSMSP=ASSSP*FFDR
         IF((DPMSP.GT.2700).OR.(DSMSP.GT.2700)) THEN
            SFTINP=.FALSE.
            RETURN
         ENDIF
C
C     DISENGAGE CLUTCHES
C
         PTIME=.TRUE.
         DLCVV=0
         DMCVV=0
      ENDIF
C
C     CHECK TO SEE IF CLUTCHES ARE SYNCHED
```

```
C
      FDR=10.48
      ALPMSP=APSSP*FDR
      ALSMSP=ASSSP*FDR
      IF((ABS(ALPMSP-APMSP).LT.400).AND.(ABS(ALSMSP-ASMSP).LT.400)) THEN
          DHCVV=0
          DLCVV=65535
          SFTINP=.FALSE.
          PTBOS=PDTBOS
          NCLTCH=.FALSE.
          FTIME=.FALSE.
      ENDIF
C
      RETURN
      END
```

```
:F2:DNSHFT.FOR

STORAGE REQUIREMENTS FOR MODULE DNSHFT:

      CODE AREA SIZE        001E3H      483D
      CONSTANT AREA SIZE    00016H       22D
      VARIABLE AREA SIZE    00008H        8D
      MAXIMUM STACK SIZE    00010H       16D
      /ERROR/               0012CH      300D
      /MOST/                000C6H      198D
      /CINOUT/              0019CH      412D
      /BITFNC/              00268H      616D
      /CALC/                0020CH      524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 13H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF DNSHFT COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
 MPILER INVOKED BY: :F1:FORT86.86  :F2:LDTREP.FOR

   1           SUBROUTINE LDTREP
   =1 *        INCLUDE(:F2:COMMON.FOR)
   2=1         INTEGER*4 RESULT,MASK(16),NAME
   3=1         INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   4=1         INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
   5=1         INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
   6=1         INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
   7=1         INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
   8=1         INTEGER*4 BIT19,BIT29,BIT39,BIT49
   9=1         INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
  10=1         INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
  11=1         INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
  12=1         INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
  13=1         INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
  14=1         INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
  15=1         INTEGER*4 DO1,DO2,DO3,DO4,DO5,DO6,DO7,DO8,DO9,DO10
  16=1         INTEGER*4 DO11,DO12,DO13,DO14,DO15,DO16,DO17,DO18,DO19,DO20
  17=1         INTEGER*4 DO21,DO22,DO23,DO24,DO25,NORMAL,POWER
  18=1         INTEGER*4 DO26,DO27,DO28,DO29,DO30
  19=1         INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
  20=1         LOGICAL*4 MSSF,TOSF,GSSF,BKNPSF,POSCSF,INBPSF
  21=1         LOGICAL*4 HEOTSF,LSCLSF,LSRPSF,LSSPSF,LPCPSF
  22=1         LOGICAL*4 LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
  23=1         LOGICAL*4 TBIT,LSPPSF,FIRESF
  24=1         LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  25=1         LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  26=1         LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
  27=1         LOGICAL*4 AEBPSF,SEWPSF,HBPVSF,PESPSF,POCSSF
  28=1         LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SPTINP
  29=1         LOGICAL*4 APBCSF,ASBCSF,BHMHSF,BHMLSF,DHMRSF,DTRNSF
  30=1         LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  31=1         LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  32=1         LOG CAL*4 MOOPCC,MLTUSF,RENTRY
  33=1         LOG CAL*4 NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
  34=1         LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  35=1         INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
  36=1         INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
  37=1         INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  38=1         INTEGER*4 TRNDIR
  39=1         INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
  40=1         INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  41=1         INTEGER*4 TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG
  42=1         INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
  43=1         INTEGER*4 CKPSLT,CKRSL1,CKRSL2
  44=1         INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  45=1         INTEGER*4 DTRST,PRMOOP
  46=1         INTEGER*4 TDELAY,TIMER,BMOOP,PBMOOP,DIGOUT(8)
  47=1         INTEGER*4 GEGFOF,PCFWNP,BLGPON
  48=1         INTEGER*4 GCTB,GRCLSE,GOTB,GROPEN,PCFWBP,BLGOFF,SLOWER
  49=1         INTEGER*4 SUBDWN,BLCVV,BHCVV,MNBPON,MBPON,TRKSTP,SURISE
  50=1         INTEGER*4 ERRDLY,DTBGS,PDTBGS,PTBGS
  51=1         INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  52=1         INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
  53=1         INTEGER*4 PTVV,STVV
  54=1         REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
   55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
   56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1          REAL*4 DPMSP,DSMSP,ODHMSP,PNTIG,AUX9,AUX10
   58=1          REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
   59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
   60=1          REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
   61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
   62=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
   63=1          REAL*4 DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
   64=1          REAL*4 PPDIS,PPDFP,APSSP,ASSSF
   65=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
   66=1          REAL*4 SPDIS,SPDFP,MAXMSP,TREP,DES
   67=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1          REAL*4 M29,M30,M31,M32,M33,M34
   74=1          REAL*4 DHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   75=1          INTEGER*4 M2,M3,M10,M11,M12
   76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1          COMMON /BITFNC/ RESULT,MASK,NAME
   79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1          COMMON /ERROR/MSSF,TSSF,GSSF,BKNPSF,PGSCSF,INDPSF
   98=1          COMMON /ERROR/ HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
   99=1          COMMON /ERROR/ LSCPSF,HPBPSF,LEOPSF,LPHLSF,LSHLSF
  100=1          COMMON /ERROR/ LSPPSF,FIRESF
  101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,POCSSF
  105=1          COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
  106=1          COMMON /ERROR/ APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
  107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  109=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
  110=1          COMMON /ERROR/ NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
  111=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  112=1          COMMON /CINOUT/ IAPBCN,IASBCN,IDHMSP IDHMSR,IPAMPR,IPFMPR
```

```
FORTRAN-86 COMPILER
:F2:LDTREP.FOR

 113=1         COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
 114=1         COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
 115=1         COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
 116=1         COMMON /CINOUT/ TEST1,TEST2,TEST3,HYSPSS,HYNSPS,ENGNON,ENGMSG
 117=1         COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
 118=1         COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
 119=1         COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
 120=1         COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDOS,PDTDOS,PTDOS
 121=1         COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
 122=1         COMMON /CINOUT/ SECFOF,PCFWNP,BLGPON
 123=1         COMMON /CINOUT/ SCTB,SRCLSE,SOTB,SROPEN,PCFWBP,BLGOFF,SLOWER
 124=1         COMMON /CINOUT/ SUBDWN,BLGVV,BHGVV,MNBPON,MBPON,TRKSTP,SURISE
 125=1         COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
 126=1         COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
 127=1         COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
 128=1         COMMON /CINOUT/ PTVV,STVV
 129=1         COMMON /CALC/ APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
 130=1         COMMON /CALC/ FDR,WUCNST
 131=1         COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
 132=1         COMMON /CALC/ DPMSP,DSMSP,DDHMSP,PNT10,AUX9,AUX10
 133=1         COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
 134=1         COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
 135=1         COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
 136=1         COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
 137=1         COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
 138=1         COMMON /CALC/ DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
 139=1         COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
 140=1         COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
 141=1         COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
 142=1         COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
 143=1         COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
 144=1         COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
 145=1         COMMON /CALC/ DHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
 146=1         COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
 147=1         COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
 148=1         COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
 149=1         COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
 150=1         COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
 151           MINUS=-1.0
 152           APPS=AENSP*1.1
 153           ASPS=AENSP*1.1
          C
          C    CALCULATE PORT MOTOR MOTREP
          C
 154           IF((APMSP/APPS).LE..49) THEN
 155               PMDIS=11.36
 156               ELSE
 157               PMDIS=6.12*APPS*0.9025/APMSP
 158           ENDIF
          C
          C    CALCULATE TORQUE
          C
 159           PMDFP=PFMTPR-PAMTPR
 160           IF(DTRST.EQ.REVRSE)
             +     PMDFP=PMDFP*MINUS
 161           IF((PAMHSF).OR.(PAMLSF).OR.(PFMHSF).OR.(PFMLSF)) PMDFP=2000.0
 162           PMTRQ=PMDIS*PMDFP/75.4
          C
```

```
      C     CALCULATE DESIRED HORSEPOWER
      C
            DPMP=PMTRQ*DPMSF/5250
            PTREFF=(ABS(DPMP/249))**0.25
            IF(PTREFF.LT.0.4) PTREFF=0.4
            REPPT=DPMP/PTREFF
      C
      C     CALCULATE STARBOARD MOTOR MOTREP
      C
            IF((ASMSP/ASPS).LE..49) THEN
                 SMDIS=11.36
                 ELSE
                 SMDIS=6.12*ASPS*0.9025/ASMSP
            ENDIF
      C
      C     CALCULATE TORQUE
      C
            SMDFP=SFMTPR-SAMTPR
            IF(DTRST.EQ.REVRSE)
           +     SMDFP=SMDFP*MINUS
            IF((SAMHSP).OR.(SAMLSP).OR.(SFMHSP).OR.(SFMLSP)) SMDFP=2000.0
            SMTRQ=SMDIS*SMDFP/75.4
      C
      C     CALCULATE DESIRED HORSEPOWER
      C
            DSMP=SMTRQ*DSMSF/5250
            STREFF=(ABS(DSMP/249))**0.25
            IF(STREFF.LT.0.4) STREFF=0.4
            REPST=DSMP/STREFF
      C
      C     REQUIRED POWER FOR TRANSMISSIONS
      C
            REPT=REPPT+REPST
      C
            RETURN
            END
```

```
FORTRAN-86 COMPILER        ERROR MESSAGES AND SUMMARY
:F2:LDTREP.FOR

STORAGE REQUIREMENTS FOR MODULE LDTREP:

    CODE AREA SIZE         003B0H        944D
    CONSTANT AREA SIZE     00036H         54D
    VARIABLE AREA SIZE     00006H          6D
    MAXIMUM STACK SIZE     00008H          8D
    /ERROR/                0012CH        300D
    /MOUT/                 000C6H        198D
    /GINOUT/               0019CH        412D
    /BITFNG/               00268H        616D
    /CALC/                 0020CH        524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 30H
FLOATING POINT OPERATIONS WERE GENERATED.
COMPILATION OF LDTREP COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
FORTRAN-86 COMPILER
:F2:PWMOUT.FOR

SERIES-III FORTRAN-86 COMPILER V1.0
MPILER INVOKED BY: :F1:FORT86.86  :F2:PWMOUT.FOR

  1          SUBROUTINE PWMOUT
   =1 $      INCLUDE(:F2:COMMON.FOR)
  2=1        INTEGER*4 RESULT,MASK(16),NAME
  3=1        INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
  4=1        INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
  5=1        INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
  6=1        INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
  7=1        INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
  8=1        INTEGER*4 BIT19,BIT29,BIT39,BIT49
  9=1        INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
 10=1        INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
 11=1        INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
 12=1        INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
 13=1        INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
 14=1        INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
 15=1        INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
 16=1        INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
 17=1        INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
 18=1        INTEGER*4 DG26,DG27,DG28,DG29,DG30
 19=1        INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
 20=1        LOGICAL*4 MSSF,TSSF,OSSF,BKNPSF,PGSCSF,INDFSF
 21=1        LOGICAL*4 HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
 22=1        LOGICAL*4 LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
 23=1        LOGICAL*4 TBIT,LSPPSF,FIRESF
 24=1        LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
 25=1        LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
 26=1        LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
 27=1        LOGICAL*4 AEJPSF,SEWPSF,HDPVSF,FEBPSF,PGCSSF
 28=1        LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
 29=1        LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
 30=1        LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
 31=1        LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
 32=1        LOGICAL*4 MOOPCC,MLTSSF,RENTRY
 33=1        LOGICAL*4 NOPAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
 34=1        LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
 35=1        INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
 36=1        INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
 37=1        INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
 38=1        INTEGER*4 TRNDIR
 39=1        INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
 40=1        INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
 41=1        INTEGER*4 TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG
 42=1        INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
 43=1        INTEGER*4 CKPSLT,CKRSL1,CKRSL2
 44=1        INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
 45=1        INTEGER*4 DTRST,PRMOOP
 46=1        INTEGER*4 TDELAY,TIMER,DMOOP,PDMOOP,DIGOUT(8)
 47=1        INTEGER*4 SECFOF,PCFWNP,DLGPON
 48=1        INTEGER*4 OCTD,ORCLSE,OOTD,OROPEN,PCFWBP,DLOOFF,SLOWER
 49=1        INTEGER*4 SUBDWN,DLCVV,DHCVV,MNSPON,MSPON,TRKSTP,SURISE
 50=1        INTEGER*4 ERRDLY,DTDGS,PDTEGS,PTDGS
 51=1        INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
 52=1        INTEGER*4 RSMS,RFMS,RSSS,RFSS,RES
 53=1        INTEGER*4 PTVV,STVV
 54=1        REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
   55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
   56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1          REAL*4 BPMSP,BSMSP,OBHMSP,PNT10,AUX9,AUX10
   58=1          REAL*4 BHWSP,BHWSR,BPWSP,BSWSP
   59=1          REAL*4 APPS,ASPS,AENSP,PMBIS,PMBFP
   60=1          REAL*4 PMTRQ,BPMP,REPPT,PTREFF,REPST
   61=1          REAL*4 APMSP,ASMSP,SMBIS,SMBFP
   62=1          REAL*4 SMTRQ,BSMP,STREFF,REPT
   63=1          REAL*4 BPFRT,PBIFP,BPPWJ,BSFRT,SBIFP,BPSWJ
   64=1          REAL*4 PPBIS,PPBFP,APSSP,ASSSP
   65=1          REAL*4 PPTRQ,BPPP,REPPP,PPMEFF,REPSP
   66=1          REAL*4 SPBIS,SPBFP,MAXMSP,TREP,DES
   67=1          REAL*4 SPTRQ,BSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1          REAL*4 K1,K2,K3,k4,K5,K6,K7,K8,K9,K10
   70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1          REAL*4 M29,M30,M31,M32,M33,M34
   74=1          REAL*4 BHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   75=1          INTEGER*4 M2,M3,M10,M11,M12
   76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1          COMMON /BITFNC/ RESULT,MASK,NAME
   79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1          COMMON /ERROR/MSSF,TSSF,OSSF,EKNFSF,POSCSF,INLPSF
   98=1          COMMON /ERROR/ HEOTSF,LSCLSF,LBRPSF,LBSPSF,LPCPSF
   99=1          COMMON /ERROR/ LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
  100=1          COMMON /ERROR/ LSPPSF,FIRESF
  101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,PGCSSF
  105=1          COMMON /ERROR/ RGLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
  106=1          COMMON /ERROR/ APBCSF,ASBCSF,BHMHSF,BHMLSF,BHMRSF,BTRNSF
  107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  109=1          COMMON /ERROR/ MOOPCD,MLTSSF,RENTRY
  110=1          COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
  111=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  112=1          COMMON /CINOUT/ IAPBCN,IASBCN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
  113=1         COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  114=1         COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  115=1         COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  116=1         COMMON /CINOUT/ TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  117=1         COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  118=1         COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  119=1         COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  120=1         COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
  121=1         COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
  122=1         COMMON /CINOUT/ SECPOF,PCPWNP,DLGPON
  123=1         COMMON /CINOUT/ DCTD,GRCLSE,DOTD,GROPEN,PCPWDP,DLGOFF,SLOWER
  124=1         COMMON /CINOUT/ SUBDWN,DLCVV,DHCVV,MNSPON,MSPON,TRKSTP,SURISE
  125=1         COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
  126=1         COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  127=1         COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
  128=1         COMMON /CINOUT/ PTVV,STVV
  129=1         COMMON /CALC/ APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  130=1         COMMON /CALC/ FDR,WJCNST
  131=1         COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  132=1         COMMON /CALC/ DPMSP,DSMSP,ODHMSP,PNTIG,AUX9,AUX10
  133=1         COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
  134=1         COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
  135=1         COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
  136=1         COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
  137=1         COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
  138=1         COMMON /CALC/ DPPRT,PDIFP,DPPWJ,DSPRT,SDIFP,DPSWJ
  139=1         COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
  140=1         COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
  141=1         COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
  142=1         COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  143=1         COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  144=1         COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  145=1         COMMON /CALC/ DHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
  146=1         COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
  147=1         COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
  148=1         COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
  149=1         COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
  150=1         COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
          C
          C     OUTPUT ENGINE GOVENOR CONTROL VOLTAGE
          C
  151           IF(DES .LE. 700.0) DES=701.1
  152           IF((M32-DES).GT.M31) DES=M32-M31
  153           IF((DTRST.EQ.PARK).OR.(DTRST.EQ.NTRAL)) DES=DHMSP+701.1
  154           IF(DHMSP.LT.1.0) DES=701.1
  155           IDES=INT(1.75*(DES-700.0))
  156           M32=DES
          C
          C     OUTPUT TRANSMISSION VALVE VOLTAGES
          C
  157           APS=AENSP*1.1
          C
  158           IF(PRMOOP.EQ.SEA) DTRST=DRIVE
  159           IF((DTRST.EQ.DRIVE).OR.(DTRST.EQ.REVRSE)) THEN
  160               IF(PRMOOP.EQ.LAND) THEN
  161                   PTVV=INT(((ALPMSP/K1)*APS**K2*((18000-PMDFP)/18000)**K3
               +        )**K4+K5)*2730
  162                   STVV=INT(((ALSMSP/K1)*APS**K2*((18000-SMDFP)/18000)**K3
          +         )**K4+K3)*2730
                    IF(ALPMSP.LT.1.0) PTVV=0
                    IF(ALSMSP.LT.1.0) STVV=0
               ELSE
                    PTVV=INT(((ALPWSP/K6)*APS**K2*((18000-PDIFF)/18000)**K5
          +         )**K7+K3)*2730
                    STVV=INT(((ALSWSP/K6)*APS**K2*((18000-SDIFF)/18000)**K5
          +         )**K7+K3)*2730
                    IF(ALPWSP.LT.1.0) PTVV=0
                    IF(ALSWSP.LT.1.0) STVV=0
               ENDIF
               IF(DTRST.EQ.DRIVE) THEN
                    PFTVV=PTVV
                    SFTVV=STVV
                    SATVV=0
                    PATVV=0
               ELSE
                    PATVV=PTVV
                    SATVV=STVV
                    PFTVV=0
                    SFTVV=0
               ENDIF
               IF((DTRNR.EQ.-1.0).AND.(PRMODP.NE.SEA)) THEN
                    IF(((DTRST.EQ.DRIVE).AND.(TRNDIR.EQ.1)).OR.
          +         ((DTRST.EQ.REVRSE).AND.(TRNDIR.EQ.-1))) THEN
                         PATVV=PTVV
                         SFTVV=STVV
                         PFTVV=0
                         SATVV=0
                    ELSE
                         PFTVV=PTVV
                         SATVV=STVV
                         PATVV=0
                         SFTVV=0
                    ENDIF
               ENDIF
          ELSE
             PFTVV=0
             SFTVV=0
             SATVV=0
             PATVV=0
          ENDIF
      C
          RETURN
          END
```

```
STORAGE REQUIREMENTS FOR MODULE PWMOUT:

      CODE AREA SIZE        00768H      1896D
      CONSTANT AREA SIZE    0001EH        30D
      VARIABLE AREA SIZE    00008H         8D
      MAXIMUM STACK SIZE    0001CH        28D
      /ERROR/               0012CH       300D
      /MOUT/                000C6H       198D
      /CINOUT/              0019CH       412D
      /BITFNC/              00268H       616D
      /CALC/                0020CH       524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 20H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF PWMOUT COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
FORTRAN-86 COMPILER
:F2:ANLGIN.FOR

SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY:  :F1:FORT86.86   :F2:ANLGIN.FOR

    1                   SUBROUTINE ANLGIN
     =1 $        INCLUDE (:F2:COMMON.FOR)
    2=1          INTEGER*4 RESULT,MASK(16),NAME
    3=1          INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
    4=1          INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
    5=1          INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
    6=1          INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
    7=1          INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
    8=1          INTEGER*4 BIT19,BIT29,BIT39,BIT49
    9=1          INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
   10=1          INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
   11=1          INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
   12=1          INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
   13=1          INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
   14=1          INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
   15=1          INTEGER*4 BG1,BG2,BG3,BG4,BG5,BG6,BG7,BG8,BG9,BG10
   16=1          INTEGER*4 BG11,BG12,BG13,BG14,BG15,BG16,BG17,BG18,BG19,BG20
   17=1          INTEGER*4 BG21,BG22,BG23,BG24,BG25,NORMAL,POWER
   18=1          INTEGER*4 BG26,BG27,BG28,BG29,BG30
   19=1          INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
   20=1          LOGICAL*4 MSGF,TSSF,GSSF,BKNPSF,PGSCSF,INBPSF
   21=1          LOGICAL*4 HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
   22=1          LOGICAL*4 LSCPSF,HFDFSF,LEOPSF,LPHLSF,LSHLSF
   23=1          LOGICAL*4 TLIT,LSPPSF,FIRESF
   24=1          LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
   25=1          LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
   26=1          LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
   27=1          LOGICAL*4 AEDPSF,SEWPSF,MDPVSF,PEDFSF,PGCSSF
   28=1          LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SPTINF
   29=1          LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
   30=1          LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
   31=1          LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
   32=1          LOGICAL*4 MODPCC,MLTSSF,RENTRY
   33=1          LOGICAL*4 NOPAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
   34=1          LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
   35=1          INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
   36=1          INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
   37=1          INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
   38=1          INTEGER*4 TRNDIR
   39=1          INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
   40=1          INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
   41=1          INTEGER*4 TEST1,TEST2,TEST3,HYSPSS,HYNDPS,ENGNOM,ENGMSG
   42=1          INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
   43=1          INTEGER*4 CKPSLT,CKRSL1,CKRSL2
   44=1          INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
   45=1          INTEGER*4 DTRST,PRMOOP
   46=1          INTEGER*4 TDELAY,TIMER,BMOOP,PBMOOP,DIGOUT(8)
   47=1          INTEGER*4 SEGFOF,PGFWNP,BLOPGN
   48=1          INTEGER*4 SGTB,GRCLSE,SGTD,GROPEN,PCFWBP,BLGOFF,SLOWER
   49=1          INTEGER*4 SUDOWN,ALCVV,BHCVV,MNBPGN,MBPGN,TRKSTP,SURISE
   50=1          INTEGER*4 ERRDLY,LTBGS,PBTBGS,PTBGS
   51=1          INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
   52=1          INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
   53=1          INTEGER*4 PTVV,STVV
   54=1          REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
  55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
  56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  57=1          REAL*4 DPMSP,DSMSP,ODHMSP,PNT10,AUX9,AUX10
  58=1          REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
  59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
  60=1          REAL*4 PMTRQ,DPMF,REPPT,PTREFF,REPST
  61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
  62=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
  63=1          REAL*4 DPFRT,PDIFF,DFPWJ,DSFRT,SDIFF,DPSWJ
  64=1          REAL*4 PPDIS,PPDFP,APSSP,ASSSP
  65=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
  66=1          REAL*4 SPDIS,SPDFP,MAXMSP,TREP,DES
  67=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
  71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
  72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
  73=1          REAL*4 M29,M30,M31,M32,M33,M34
  74=1          REAL*4 DHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
  75=1          INTEGER*4 M2,M3,M10,M11,M12
  76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
  77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
  78=1          COMMON /BITFNC/ RESULT,MASK,NAME
  79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
  80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
  81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
  82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
  83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
  84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
  85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
  86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
  87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
  88=1          COMMON /BIT  C/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
  89=1          COMMON /BIT  C/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
  90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
  91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
  92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
  93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
  94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
  95=1          COMMON /BT  NC/ DG26,DG27,DG28,DG29,DG30
  96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
  97=1          COMMON /ERROR/MSSF,TSSF,GSSF,BKNPSF,PGSCSF,INDPSF
  98=1          COMMON /ERROR/ MEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
  99=1          COMMON /ERROR/ LSCPSF,MFDPSF,LEOPSF,LPHLSF,LSHLSF
 100=1          COMMON /ERROR/ LSPPSF,FIRESF
 101=1          COMMON /ERROR/ F1SCSF,F2SCSF,F3SCSF,F4SCSF,F5SCSF
 102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
 103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
 104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HDPVSF,FEBPSF,PGCSSF
 105=1          COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
 106=1          COMMON /ERROR/ APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
 107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
 108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
 109=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
 110=1          COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
 111=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
 112=1          COMMON /CINOUT/ IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
```

```
  113=1        COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  114=1        COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  115=1        COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  116=1        COMMON /CINOUT/ TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  117=1        COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  118=1        COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  119=1        COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  120=1        COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,BTDOS,PDTDOS,PTDOS
  121=1        COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
  122=1        COMMON /CINOUT/ SECFOF,PCFWNP,BLGPON
  123=1        COMMON /CINOUT/ OSTD,GRCLSE,OOTD,GROPEN,PCFWDP,BLGOFF,SLOWER
  124=1        COMMON /CINOUT/ SUDOWN,BLCVV,DHCVV,MNDPON,MDPON,TRKSTP,SURISE
  125=1        COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
  126=1        COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  127=1        COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
  128=1        COMMON /CINOUT/ PTVV,STVV
  129=1        COMMON /CALC/ APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  130=1        COMMON /CALC/ FDR,WJCNST
  131=1        COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  132=1        COMMON /CALC/ DPMSP,DSMSP,ODHMSP,PNT10,AUX9,AUX10
  133=1        COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
  134=1        COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
  135=1        COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
  136=1        COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
  137=1        COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
  138=1        COMMON /CALC/ DPFRT,PDIFF,DPPWJ,DSFRT,SDIFF,DPSWJ
  139=1        COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
  140=1        COMMON /CALC/ PPTRQ,DFPP,REPPP,PPMEFF,REPSP
  141=1        COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREP,DES
  142=1        COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  143=1        COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  144=1        COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  145=1        COMMON /CALC/ BHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
  146=1        COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
  147=1        COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
  148=1        COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
  149=1        COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
  150=1        COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
           C   DIMENSION BHMSPD(10),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
           C
           C   PASS THE 12 LSB'S TO READ OUTPUTS
           C
  151          RSMS=(IASMSP)
  152          RPMS=(IAPMSP)
  153          RSSS=(IASSSP)
  154          RPSS=(IAPSSP)
  155          RES=(IAENSP)
           C
           C   CONVERT BUCKET ANGLES
           C
  156          APBCAN=0.0265*FLOAT(IAPBCN)-7.2
  157          IF((APBCAN.GE.87).AND.(APBCAN.LE.93)) APBCAN=90
  158          IF((APBCAN.GE.-3).AND.(APBCAN.LE.3)) APBCAN=0
  159          IF((APBCAN.LE.-3).OR.(APBCAN.GE.93)) THEN
  160             APBCAN=90.0
  161             APBCSF=.TRUE.
  162          ENDIF
           C
```

```
        ASBCAN=0.0255*FLOAT(IASBCN)-7.2
        IF((ASBCAN.GE.87).AND.(ASBCAN.LE.93)) ASBCAN=90
        IF((ASBCAN.GE.-3).AND.(ASBCAN.LE.3)) ASBCAN=0
        IF((ASBCAN.LE.-3).OR.(ASBCAN.GE.93)) THEN
            ASBCAN=90.0
            ASBCSF=.TRUE.
        ENDIF
   C
   C    CONVERT DESIRED HIGH MOTOR SPEED AND DESIRED HIGH MOTOR SPEED RATIO
   C
        M10=M10+1
        M11=M11+1
        DHMSP=0.0
        DHMSPB(M11)=1.077*FLOAT(IDHMSP)-705.6
        DO 10 J=1,M12
            DHMSP=DHMSP+DHMSPB(J)
   10   CONTINUE
        DHMSP=DHMSP/M13
        IF((DHMSP.GT.3000).AND.(DHMSP.LE.3600)) DHMSP=3000
        IF(DHMSP.GT.3600) THEN
            DHMSP=0
            DHMHSF=.TRUE.
        ENDIF
        IF((DHMSP.LT.0).AND.(DHMSP.GT.-570)) DHMSP=0
        IF(DHMSP.LE.-570) THEN
            DHMSP=0
            DHMLSF=.TRUE.
        ENDIF
        DHWSP=DHMSP
        IF(M11.EQ.M12) M11=0
   C
        DHMSR=.0003591*FLOAT(IDHMSR)-0.2352
        IF((DHMSR.GT.1).AND.(DHMSR.LE.1.2)) DHMSR=1
        IF((DHMSR.GE.-0.2).AND.(DHMSR.LT.0)) DHMSR=0
        IF((DHMSR.GT.1.2).OR.(DHMSR.LT.-0.2)) THEN
            DHMSR=0
            DHMRSF=.TRUE.
        ENDIF
        DHWSR=DHMSR
        IF((DHMSR.GT.0.9).AND.(LBRPSF)) DKNPSF=.TRUE.
        IF((DHMSR.LT.0.9).AND.(LSRPSF)) INSPSF=.TRUE.
   C
   C    DESIRED TURN RATIO
   C
        IF(IDTRNR.LT.1540) THEN
            TRNDIR=1
            ELSE
            TRNDIR=-1
        ENDIF
   C
        DTRNR=TRNDIR*.00097*FLOAT(IDTRNR)+INTCPT(TRNDIR)
        IF(DTRNR.GE.1.0) DTRNR=1.0
        IF((DTRNR.LE.0.05).AND.(DTRNR.GT.-0.113)) DTRNR=0.0
        IF(DTRNR.LE.-0.33) THEN
            DTRNR=1.0
            DTRNSF=.TRUE.
        ENDIF
        IF((DTRNR.LE.-0.113).AND.(DTRNR.GT.-0.33)) DTRNR=-1.0
```

```
C
C      CONVERT MOTOR PRESSURES
C
       PAMTPR=0.0
       PAMPR(M10)=3.4686*FLOAT(IPAMPR)-2500.86
       DO 20 J=1,M35
           PAMTPR=PAMTPR+PAMPR(J)
 20    CONTINUE
       PAMTPR=PAMTPR/M34
       IF(PAMTPR.GT.12000) PAMHSF=.TRUE.
       IF(PAMTPR.LE. -50) PAMLSF=.TRUE.
C
       PFMTPR=0.0
       PFMPR(M10)=3.4686*FLOAT(IPFMPR)-2500.86
       DO 30 J=1,M35
           PFMTPR=PFMTPR+PFMPR(J)
 30    CONTINUE
       PFMTPR=PFMTPR/M34
       IF(PFMTPR.GT.12000) PFMHSF=.TRUE.
       IF(PFMTPR.LE. -50) PFMLSF=.TRUE.
C
       SFMTPR=0.0
       SFMPR(M10)=3.4686*FLOAT(ISFMPR)-2500.86
       DO 40 J=1,M35
           SFMTPR=SFMTPR+SFMPR(J)
 40    CONTINUE
       SFMTPR=SFMTPR/M34
       IF(SFMTPR.GT.12000) SFMHSF=.TRUE.
       IF(SFMTPR.LE.-50) SFMLSF=.TRUE.
C
       SAMTPR=0.0
       SAMPR(M10)=3.4686*FLOAT(ISAMPR)-2500.86
       DO 50 J=1,M35
           SAMTPR=SAMTPR+SAMPR(J)
 50    CONTINUE
       SAMTPR=SAMTPR/M34
       IF(SAMTPR.GT.12000) SAMHSF=.TRUE.
       IF(SAMTPR.LE.-50) SAMLSF=.TRUE.
       IF(M10.EQ.M35) M10=0
C
C      CONVERT ENGINE AND HYDRAULIC TEMPERATURES
C
       ENCTEM=.0624*FLOAT(IENCTM)-12.99
       IF((ENCTEM.LT.-10).OR.(ENCTEM.GT.210)) THEN
            ENCTEM=240
            ENCTSF=.TRUE.
       ENDIF
C
       PHOTEM=.0624*FLOAT(IPHOTM)-12.99
       IF((PHOTEM.LT. -10).OR.(PHOTEM.GT.180)) THEN
            PHOTEM=240
            PHOTSF=.TRUE.
       ENDIF
C
       SHOTEM=.0624*FLOAT(ISHOTM)-12.99
       IF((SHOTEM.LT. -10).OR.(SHOTEM.GT.180)) THEN
            SHOTEM=240
            SHOTSF=.TRUE.
 260           ENDIF
      C
 261           RETURN
 262           END
```

```
:F2:ANLGIN.FOR

STORAGE REQUIREMENTS FOR MODULE ANLGIN:

      CODE AREA SIZE           00EA0H       3744D
      CONSTANT AREA SIZE       00072H        114D
      VARIABLE AREA SIZE       00006H          6D
      MAXIMUM STACK SIZE       0001EH         30D
      /ERROR/                  0012CH        300D
      /MOUT/                   000C6H        198D
      /GINOUT/                 0019CH        412D
      /UITFNC/                 00268H        616D
      /CALC/                   0020CH        524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 94H
FLOATING POINT OPERATIONS WERE GENERATED.
COMPILATION OF ANLGIN COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY: :F1:FORT86.86  :F2:DEMOS.FOR

    1            SUBROUTINE DEMOS
   =1  $         INCLUDE (:F2:COMMON.FOR)
    2=1          INTEGER*4 RESULT,MASK(16),NAME
    3=1          INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
    4=1          INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
    5=1          INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
    6=1          INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
    7=1          INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
    8=1          INTEGER*4 BIT19,BIT29,BIT39,BIT49
    9=1          INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
   10=1          INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
   11=1          INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
   12=1          INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
   13=1          INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
   14=1          INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
   15=1          INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   16=1          INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   17=1          INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   18=1          INTEGER*4 DG26,DG27,DG28,DG29,DG30
   19=1          INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
   20=1          LOGICAL*4 MSSF,TSSF,OSSF,DKNPSF,PGSCSF,INDPSF
   21=1          LOGICAL*4 MEOTSF,L3CLSF,L3RPSF,L3SPSF,LPCPSF
   22=1          LOGICAL*4 LSCPSF,HPBPSF,LEOPSF,LPHLSF,LSHLSF
   23=1          LOGICAL*4 TBIT,LSPPSF,FIRESF
   24=1          LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
   25=1          LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
   26=1          LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
   27=1          LOGICAL*4 AEBPSF,SEWPSF,HDPVSF,FEBPSF,FGCSSF
   28=1          LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
   29=1          LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
   30=1          LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
   31=1          LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
   32=1          LOGICAL*4 MOOPCC,MLTSSF,RENTRY
   33=1          LOGICAL*4 NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
   34=1          LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
   35=1          INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
   36=1          INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
   37=1          INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
   38=1          INTEGER*4 TRNDIR
   39=1          INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
   40=1          INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
   41=1          INTEGER*4 TEST1,TEST2,TEST3,HYBPSS,HYNDPS,ENGNON,ENGMSG
   42=1          INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
   43=1          INTEGER*4 CKPSLT,CKRSL1,CKRSL2
   44=1          INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP IASSSP,IAPWSP,IASWSP
   45=1          INTEGER*4 DTRST,PRMOOP
   46=1          INTEGER*4 TDECAY,TIMER,DMOOP,PDMOOP,DIGOUT(8)
   47=1          INTEGER*4 SECPOF,PCFWNP,SLOPON
   48=1          INTEGER*4 OCTD,ORCLSE,OOTD,GROPEN,PCFWDP,DLGOFF,SLOWER
   49=1          INTEGER*4 SUDOWN,DLCVV,DHCVV,MNSPON,MSPON,TRKSTP,SURISE
   50=1          INTEGER*4 ERRDLY,DTDOS,PDTDOS,FTDOS
   51=1          INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
   52=1          INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
   53=1          INTEGER*4 PTVV,STVV
   54=1          REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
          REAL*4 FDR,WJCNST,INTCPT(-1:1)
          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
          REAL*4 BPMSP,BSMSP,DBHMSP,PNT10,AUX9,AUX10
          REAL*4 BHWSP,BHWSR,BPWSP,BSWSP
          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
          REAL*4 PMTRQ,BPMP,REPPT,PTREFF,REPST
          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
          REAL*4 SMTRQ,BSMP,STREFF,REPT
          REAL*4 BPFRT,PBIFP,BPPWJ,BSFRT,SBIFP,BPSNU
          REAL*4 PPDIS,PPDFP,APSSP,ASSSP
          REAL*4 PPTRQ,BPPP,REPPP,PPMEFF,REPSP
          REAL*4 SPDIS,SPDFP,MAXMSP,TREP,BES
          REAL*4 SPTRQ,BSPP,SPMEFF,REPP,ALPMSP,ALSMSP
          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
          REAL*4 M1,M4,M5,M6,M7,M8,M9
          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
          REAL*4 M29,M30,M31,M32,M33,M34
          REAL*4 BHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
          INTEGER*4 M2,M3,M10,M11,M12
          INTEGER*4 M35,M36,M37,M38,M39,M40
          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
          COMMON /BITFNC/ RESULT,MASK,NAME
          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
          COMMON /ERROR/MSSF,TSSF,OSSF,DKNPSF,PSSCSF,INDPSF
          COMMON /ERROR/ HEDTSF,LBCLSF,LDRPSF,LBSPSF,LPCPSF
          COMMON /ERROR/ LSCPSF,HFEPSF,LEOPSF,LPHLSF,LSHLSF
          COMMON /ERROR/ LSPPSF,FIRESF
          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,PGOSSF
          COMMON /ERROR/ RGLOSF,LGTRNS,SLTRNS,INITIAL,SFTINP
          COMMON /ERROR/ APBCSF,ASBCSF,BHMHSF,BHMLSF,BHMRSF,BTRNSF
          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
          COMMON /ERROR/ MODPCC,MLTSSF,RENTRY
          COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
          COMMON /CINOUT/ IAPSCN,IASSCN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
```

```
 113=1         COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
 114=1         COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
 115=1         COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
 116=1         COMMON /CINOUT/ TEST1,TEST2,TEST3,HYBPSS,HYNOPS,ENGNON,ENGMSG
 117=1         COMMON /CINOUT/ CKPCMP,CKRAMP,CKPLST,CKRLST
 118=1         COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
 119=1         COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
 120=1         COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
 121=1         COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
 122=1         COMMON /CINOUT/ SECFOF,PCFWNP,BLGPON
 123=1         COMMON /CINOUT/ GCTD,GRCLSE,GOTD,GROPEN,PCFWBP,BLGOFF,SLOWER
 124=1         COMMON /CINOUT/ SUDOWN,DLCVV,DHCVV,MNBPON,MBPON,TRKSTP,SURISE
 125=1         COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
 126=1         COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
 127=1         COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
 128=1         COMMON /CINOUT/ PTVV,STVV
 129=1         COMMON /CALC/ APECAN,ASECAN,DHMSP,DHMSR,DTRNR,PAMTPR
 130=1         COMMON /CALC/ PDR,WJCNST
 131=1         COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
 132=1         COMMON /CALC/ DPMSP,DSMSP,ODHMSP,PNT10,AUX9,AUX10
 133=1         COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
 134=1         COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
 135=1         COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
 136=1         COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
 137=1         COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
 138=1         COMMON /CALC/ DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
 139=1         COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
 140=1         COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
 141=1         COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREF,DES
 142=1         COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
 143=1         COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
 144=1         COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
 145=1         COMMON /CALC/ DHMSP1,PAMPR,SAMPR,PFMPR,SFMPR
 146=1         COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
 147=1         COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
 148=1         COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
 149=1         COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
 150=1         COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
           C
           C   DETERMINE DESIRED MOTOR SPEEDS ACCORDING TO PRESENT MODE OF OPER.
           C
           C   LANDBORNE
           C
 151           IF(PTDGS.EQ.HIGH) THEN
 152                PNT10=M29
 153                ELSE
 154                PNT10=M28
 155           ENDIF
           C
 156           IF(PRMOOP.EQ.LAND) THEN
 157                AMSP=MAX(APMSP,ASMSP)
 158                IF(DHMSP+300.LT.AMSP) THEN
 159                     IF(ODHMSP.GT.AMSP) THEN
 160                          DHMSP=AMSP-PNT10
 161                          ELSE
 162                          DHMSP=ODHMSP-PNT10
 163                     ENDIF
 164                ENDIF
```

```
          DHMSP=DHMSP*DHMSR
          ODHMSP=DHMSP
          IF(TRNDIR.EQ.1) THEN
              DPMSP=DHMSP*DTRNR
              DSMSP=DHMSP
              ELSE
              DPMSP=DHMSP
              DSMSP=DHMSP*DTRNR
          ENDIF
      ENDIF
C
C     TRANSITION (DESIRED WATERJET SPEEDS)
C
      IF(PRMODP.EQ.TRNSTN) THEN
          AWSP=MAX(APWSP,ASWSP)
          IF(DHWSP+900.LT.AWSP) THEN
              IF(ODHWSP.GT.AWSP) THEN
                  DHWSP=AWSP-PNT16
                  ELSE
                  DHWSP=ODHWSP-PNT16
              ENDIF
          ENDIF
          DHWSP=DHWSP*DHMSR
          ODHWSP=DHWSP
          IF(TRNDIR.EQ.1) THEN
              DPWSP=DHWSP*DTRNR
              DSWSP=DHWSP
              ELSE
              DPWSP=DHWSP
              DSWSP=DHWSP*DTRNR
          ENDIF
      ENDIF
C
C     SEABORNE
C
      IF(PRMODP.EQ.SEA) THEN
          DHWSP=DHWSP*DHWSR
          DPWSP=DHWSP
          DSWSP=DHWSP
      ENDIF
C
      RETURN
      END
```

```
FORTRAN-86 COMPILER          ERROR MESSAGES AND SUMMARY
:F2:DEMOS.FOR

STORAGE REQUIREMENTS FOR MODULE DEMOS:

      CODE AREA SIZE          00345H        837D
      CONSTANT AREA SIZE      0000EH         14D
      VARIABLE AREA SIZE      00018H         24D
      MAXIMUM STACK SIZE      00012H         18D
      /ERROR/                 0012CH        300D
      /MOUT/                  000C6H        198D
      /CINOUT/                0019CH        412D
      /BITFNC/                00268H        616D
      /CALC/                  0020CH        524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 10H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF DEMOS COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY: :F1:FORT86.86  :F2:ECAN.FOR

     1           SUBROUTINE ECAN
    *1  *        INCLUDE (:F2:COMMON.FOR)
    2=1          INTEGER*4 RESULT,MASK(16),NAME
    3=1          INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
    4=1          INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
    5=1          INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
    6=1          INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
    7=1          INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
    8=1          INTEGER*4 BIT19,BIT29,BIT39,BIT49
    9=1          INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
   10=1          INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
   11=1          INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
   12=1          INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
   13=1          INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
   14=1          INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
   15=1          INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   16=1          INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   17=1          INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   18=1          INTEGER*4 DG26,DG27,DG28,DG29,DG30
   19=1          INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
   20=1          LOGICAL*4 MSSF,TSSF,GSSF,BKNPSF,PGSCSF,INDFSF
   21=1          LOGICAL*4 HEOTSF,LSCLSF,LSRPSF,LSSPSF,LPCPSF
   22=1          LOGICAL*4 LSCPSF,HFBFSF,LEOFSF,LFHLSF,LSHLSF
   23=1          LOGICAL*4 TBIT,LSPPSF,FIRESF
   24=1          LOGICAL*4 F1SCSF,F2SCSF,F3SCSF,F4SCSF,F5SCSF
   25=1          LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
   26=1          LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
   27=1          LOGICAL*4 AEDPSF,SEWPSF,MDPVSF,FESPSF,PGCSSF
   28=1          LOGICAL*4 ACLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
   29=1          LOGICAL*4 APSCSF,ASSCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
   30=1          LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
   31=1          LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
   32=1          LOGICAL*4 MUOPCC,MLTSSF,RENTRY
   33=1          LOGICAL*4 NOFAN,F40PM,F80PM,F120PM,PTIME,NCLTCH
   34=1          LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
   35=1          INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
   36=1          INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
   37=1          INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
   38=1          INTEGER*4 TRNDIR
   39=1          INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
   40=1          INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
   41=1          INTEGER*4 TEST1,TEST2,TEST3,HYDPSS,HYNBPS,ENGNON,ENGMSG
   42=1          INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
   43=1          INTEGER*4 CKPSLT,CKRSLT,CKRSL2
   44=1          INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
   45=1          INTEGER*4 DTRST,FRMOOP
   46=1          INTEGER*4 TDELAY,TIMER,BMOOP,PBMOOP,DIGOUT(8)
   47=1          INTEGER*4 GEGFOF,PCFWNP,BLOPON
   48=1          INTEGER*4 GCTB,GRCLSE,SOTB,GROPEN,PCFWBP,BLOOFF,SLOWER
   49=1          INTEGER*4 SUBOWN,BLEVV,BHEVV,MNDPON,MBPON,TRKSTP,SURISE
   50=1          INTEGER*4 ERRDLY,DTBSS,PBTBSS,PTBSS
   51=1          INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
   52=1          INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
   53=1          INTEGER*4 PTVV,STVV
   54=1          REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
  55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
  56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  57=1          REAL*4 DPMSP,DSMSP,ODMMSP,PNT16,AUX9,AUX10
  58=1          REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
  59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
  60=1          REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
  61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFP
  62=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
  63=1          REAL*4 DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
  64=1          REAL*4 PPDIS,PPDFP,APSSP,ASSSP
  65=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
  66=1          REAL*4 SPDIS,SPDFP,MAXMSP,TREP,DES
  67=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
  71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
  72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
  73=1          REAL*4 M29,M30,M31,M32,M33,M34
  74=1          REAL*4 DHMSPD(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
  75=1          INTEGER*4 M2,M3,M10,M11,M12
  76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
  77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
  78=1          COMMON /BITFNC/ RESULT,MASK,NAME
  79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
  80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
  81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
  82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
  83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
  84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
  85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
  86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
  87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
  88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
  89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
  90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
  91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
  92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
  93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
  94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
  95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
  96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
  97=1          COMMON /ERROR/MSSF,TSSF,GSSF,DKNPSF,PGSCSF,INDPSF
  98=1          COMMON /ERROR/ HEOTSF,LDCLSF,LDRPSF,LBSPSF,LPCPSF
  99=1          COMMON /ERROR/ LSCPSF,HFDPSF,LEOPSF,LPHLSF,LSHLSF
 100=1          COMMON /ERROR/ LSPPSF,FIRESF
 101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
 102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
 103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
 104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,POCSSF
 105=1          COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
 106=1          COMMON /ERROR/ APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
 107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
 108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
 109=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
 110=1          COMMON /ERROR/ NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
 111=1          COMMON /ERROR/ AENSP,APSSP,APMSP,ASSSP,ASMSP,APWSP,ASWSP
 112=1          COMMON /CINOUT/ IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
```

```
  113=1          COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  114=1          COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  115=1          COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  116=1          COMMON /CINOUT/ TEST1,TEST2,TEST3,HYBPSS,HYNBPS,ENGNON,ENGMSG
  117=1          COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  118=1          COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  119=1          COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  120=1          COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
  121=1          COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
  122=1          COMMON /CINOUT/ SECFOF,PCFWNP,BLSPON
  123=1          COMMON /CINOUT/ SSTD,SRCLSE,SOTD,SROPEN,PCFWBP,BLSOFF,SLOWER
  124=1          COMMON /CINOUT/ SUBOWN,BLCVV,BHCVV,MNBPON,MBPON,TRKSTP,SURISE
  125=1          COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
  126=1          COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  127=1          COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
  128=1          COMMON /CINOUT/ PTVV,STVV
  129=1          COMMON /CALC/ APECAN,ASDCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  130=1          COMMON /CALC/ FDR,WJCNST
  131=1          COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  132=1          COMMON /CALC/ DPMSP,DSMSP,ODHMSP,PNT10,AUX9,AUX10
  133=1          COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
  134=1          COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDPP
  135=1          COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
  136=1          COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDPP
  137=1          COMMON /CALC/ SMTRQ,DSMP,STREFP,REPT
  138=1          COMMON /CALC/ DPPRT,PDIFF,DPPWJ,DSPRT,SDIFF,DPSWJ
  139=1          COMMON /CALC/ PPDIS,PPDPP,APSSP,ASSSP
  140=1          COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
  141=1          COMMON /CALC/ SPDIS,SPDPP,MAXMSP,TREP,DES
  142=1          COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REFP,ALPMSP,ALSMSP
  143=1          COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  144=1          COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  145=1          COMMON /CALC/ DHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
  146=1          COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
  147=1          COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
  148=1          COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
  149=1          COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
  150=1          COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
           C
           C     LOGIC TO TURN ON COOLING FANS
           C
  151            IF((PHOTEM.GT.180).OR.(SHOTEM.GT.180).OR.(ENCTEM.GT.210)) THEN
  152                F12GPM=.TRUE.
  153            ELSE IF((PHOTEM.GT.165).OR.(SHOTEM.GT.165).OR.(ENCTEM.GT.205))
                +THEN
  154                F8GPM=.TRUE.
  155            ELSE IF((PHOTEM.GT.140).OR.(SHOTEM.GT.140).OR.(ENCTEM.GT.200))
                +THEN
  156                F4GPM=.TRUE.
  157            ELSE
  158                NOFAN=.TRUE.
  159            ENDIF
  160            DIGOUT(1)=DIGOUT(1).AND.D[illegible]
           C
           C     TURN ON FANS AND CALC AUX POWER USAGE
           C
  161            IF(RMOOP.EQ.LAND) THEN
  162                IF(NOFAN) AUXPOW=0.11961*AENSP-0.81056E-04*AENSP**2
```

```
     +    +0.13165E-07*AENSP**3-0.94209E-12*AENSP**4-33.683
          IF(F4GPM) THEN
          AUXPOW=0.62975E-01*AENSP-0.26446E-05*AENSP**2
     +    -0.99716E-08*AENSP**3+0.22257E-11*AENSP**4-15.593
          DIGOUT(1)=DIGOUT(1).OR.DG21
          ENDIF
          IF(F8GPM) THEN
C         AUXPOW=0.62975E-01*AENSP-0.26446E-05*AENSP**2
C    +    -0.99716E-08*AENSP**3-0.22257E-11*AENSP**4-15.593
          AUXPOW=0.11461*AENSP-0.68885E-04*AENSP**2
     +    +0.24006E-07*AENSP**3-0.34259E-11*AENSP**4-29.471
          DIGOUT(1)=DIGOUT(1).OR.DG22
          ENDIF
          IF(F12GPM) THEN
C         AUXPOW=0.62975E-01*AENSP-0.26446E-05*AENSP**2
C    +    -0.99716E-08*AENSP**3-0.22257E-11*AENSP**4-15.593
          AUXPOW=0.1046*AENSP-0.55165E-04*AENSP**2
     +    +0.16275E-07*AENSP**3-0.19096E-11*AENSP**4-26.931
          DIGOUT(1)=DIGOUT(1).OR.DG23
          ENDIF
      ENDIF
C
      IF(.NOT.(NOFAN)) DIGOUT(2)=DIGOUT(2).OR.BIT39
      NOFAN=.FALSE.
      F4GPM=.FALSE.
      F8GPM=.FALSE.
      F12GPM=.FALSE.
C
      IF(PRMODP.NE.LAND) THEN
          AUXPOW=0.62074E-02*AENSP+.22895E-04*AENSP**2
     +    -0.91625E-08*AENSP**3-0.88974E-12*AENSP**4+4.3816
          DIGOUT(1)=DIGOUT(1).OR.DG21
      ENDIF
C
C     CALC DESIRED ENGINE SPEED
C
      TREP=AUXPOW*REPT
      DES=21.975*TREP-0.13364*TREP**2+0.27021E-03*TREP**3+568.02
      DES=DES*M33
      IF(DES.GT.2800.0) DES=2800.0
C
C     SET BUCKETS TO DESIRED ANGLE
C
          DANGLE=DTRNK
          IF(DANGLE.LT.0) DANGLE=0
          IF((PRMODP.EQ.SEA).AND.(DTRST.EQ.REVRSE)) THEN
          DANGLE=DANGLE*90
          IF(APBCSF) APBCAN=0.0
          IF(ASBCSF) ASBCAN=0.0
          STOWED=90.0
          ELSE
          DANGLE=90.0-DANGLE*90.0
          STOWED=0.0
          ENDIF
          IF(PRMODP.EQ.LAND)THEN
            DANGLE=90.0
             STOWED=90.0
            ENDIF
```

```
         IF(TRNDIR.EQ.1) THEN
         DIGOUT(3)=DIGOUT(3).AND.DG24
         DIGOUT(1)=DIGOUT(1).AND.DG27
         IF((BANGLE-APBCAN).GT.4.5) DIGOUT(3)=DIGOUT(3).OR.DG25
         IF((BANGLE-APBCAN).LT.-4.5) DIGOUT(3)=DIGOUT(3).OR.DG26
         IF((STOWED-ASBCAN).LT.-4.5) DIGOUT(1)=DIGOUT(1).OR.DG29
         IF((STOWED-ASBCAN).GT.4.5) DIGOUT(1)=DIGOUT(1).OR.DG28
         ELSE
         DIGOUT(1)=DIGOUT(1).AND.DG27
         DIGOUT(3)=DIGOUT(3).AND.DG24
         IF((BANGLE-ASBCAN).GT.4.5) DIGOUT(1)=DIGOUT(1).OR.DG28
         IF((BANGLE-ASBCAN).LT.-4.5) DIGOUT(1)=DIGOUT(1).OR.DG29
         IF((STOWED-APBCAN).LT.-4.5) DIGOUT(3)=DIGOUT(3).OR.DG26
         IF((STOWED-APBCAN).GT.4.5) DIGOUT(3)=DIGOUT(3).OR.DG25
         ENDIF
         RETURN
      END
```

```
FORTRAN-86 COMPILER       ERROR MESSAGES AND SUMMARY
:F2:ECAN.FOR

STORAGE REQUIREMENTS FOR MODULE ECAN:

      CODE AREA SIZE          009E4H        2532D
      CONSTANT AREA SIZE      000AAH         170D
      VARIABLE AREA SIZE      0000CH          12D
      MAXIMUM STACK SIZE      00012H          18D
      /ERROR/                 0012CH         300D
      /MOUT/                  000C6H         198D
      /CINOUT/                0019CH         412D
      /BITFNC/                00268H         616D
      /CALC/                  0020CH         524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 0ACH
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF ECAN COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
FORTRAN-86 COMPILER        13:30:00 11 DEC.85
:F2:FREQIN.FOR

SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY:  :F1:FORT86.86   :F2:FREQIN.FOR  CODE TITLE(13:30:00 11 DEC.85)

   1            SUBROUTINE FREQIN
   =1 #         INCLUDE (:F2:COMMON.FOR)
   2=1          INTEGER*4 RESULT,MASK(16),NAME
   3=1          INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   4=1          INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
   5=1          INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
   6=1          INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
   7=1          INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
   8=1          INTEGER*4 BIT19,BIT29,BIT39,BIT49
   9=1          INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
  10=1          INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
  11=1          INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
  12=1          INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
  13=1          INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
  14=1          INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
  15=1          INTEGER*4 DB1,DB2,DB3,DB4,DB5,DB6,DB7,DB8,DB9,DB10
  16=1          INTEGER*4 DB11,DB12,DB13,DB14,DB15,DB16,DB17,DB18,DB19,DB20
  17=1          INTEGER*4 DB21,DB22,DB23,DB24,DB25,NORMAL,POWER
  18=1          INTEGER*4 DB26,DB27,DB28,DB29,DB30
  19=1          INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
  20=1          LOGICAL*4 MSSF,TSSF,GSSF,BKNFSF,PGSCSF,INBFSF
  21=1          LOGICAL*4 HEOTSF,LDCLSF,LDRPSF,LDSPSF,LPCPSF
  22=1          LOGICAL*4 LSCPSF,HPDPSF,LEOPSF,LPHLSF,LSHLSF
  23=1          LOGICAL*4 TBIT,LSPPSF,FIRESF
  24=1          LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  25=1          LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  26=1          LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
  27=1          LOGICAL*4 AEDPSF,SEWPSF,HDPVSF,FEDPSF,FGCSSF
  28=1          LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INTIAL,SPTINP
  29=1          LOGICAL*4 APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
  30=1          LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  31=1          LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  32=1          LOGICAL*4 MODPCC,MLTSSF,RENTRY
  33=1          LOGICAL*4 NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
  34=1          LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  35=1          INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
  36=1          INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
  37=1          INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  38=1          INTEGER*4 TRNDIR
  39=1          INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
  40=           INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL REVRSE,DRIVE,HIGH,LOW
  41=1          INTEGER*4 TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  42=1          INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLS
  43=1          INTEGER*4 CKPSLT,CKRSL1,CKRSL2
  44=1          INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  45=1          INTEGER*4 DTRST,PRMODP
  46=1          INTEGER*4 TDELAY,TIMER,BMODP,PBMODP,DIGOUT(8)
  47=1          INTEGER*4 SEGFOF,PSFUNP,BLSPON
  48=1          INTEGER*4 GGTB,GRELSE,GGTD,GROPEN,P2FWBP,BLGOFF,SLOWER
  49=1          INTEGER*4 SUBOWN,BLCVV,BHCVV,MNLPON,MBPON,TRKSTP,SURISE
  50=1          INTEGER*4 ERRDLY,DTBGS,PDTBGS,PTBGS
    =1          INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  52=1          INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
  53=1          INTEGER*4 PTVV,STVV
  54=1          REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
```

```
   55=1         REAL*4 FDR,WJCNST,INTCPT(-1:1)
   56=1         REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1         REAL*4 DPMSP,DSMSP,DDHMSP,PNTIG,AUX9,AUX10
   58=1         REAL*4 DHWSP,DHWSR,DPWSP,DSWSP
   59=1         REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFP
   60=1         REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
   61=1         REAL*4 APMSP,ASMSP,SMDIS,SMDFP
   62=1         REAL*4 SMTRQ,DSMP,STREFF,REPT
   63=1         REAL*4 DPFRT,FDIFP,DPFWJ,DSFRT,SDIFP,DFSWJ
   64=1         REAL*4 PPDIS,PPDFP,APSSP,ASSSP
   65=1         REAL*4 PPTRQ,DPPP,REFPP,PPMEFF,REPSP
   66=1         REAL*4 SPDIS,SPDFP,MAXMSP,TREP,DES
   67=1         REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1         REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1         REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   70=1         REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1         REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1         REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1         REAL*4 M29,M30,M31,M32,M33,M34
   74=1         REAL*4 DHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   75=1         INTEGER*4 M2,M3,M10,M11,M12
   76=1         INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1         INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1         COMMON /BITFNC/ RESULT,MASK,NAME
   79=1         COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1         COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1         COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1         COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1         COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1         COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1         COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1         COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1         COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1         COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1         COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1         COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1         COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1         COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1         COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1         COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1         COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1         COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1         COMMON /ERROR/MSSF,TSSF,GSSF,BKNPSF,PGSCSF,INDPSF
   98=1         COMMON /ERROR/ MEOTSF,L3CLSF,L3RPSF,L3SPSF,LPCPSF
   99=1         COMMON /ERROR/ LSCFSF,HPDFSF,LEOPSF,LPHLSF,LSHLSF
  100=1         COMMON /ERROR/ LSPPSF,FIRESF
  101=1         COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  102=1         COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  103=1         COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  104=1         COMMON /ERROR/ AESPSF,SEWPSF,HDPVSF,PEDPSF,FOCSSF
  105=1         COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINF
  106=1         COMMON /ERROR/ APBCSF,ASBCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
  107=1         COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  108=1         COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  109=1         COMMON /ERROR/ MODFCC,MLTSSF,RENTRY
  110=1         COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
  111=1         COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  112=1         COMMON /CINOUT/ IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,IPFMPR
  113=1        COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  114=1        COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  115=1        COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  116=1        COMMON /CINOUT/ TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  117=1        COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  118=1        COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  119=1        COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  120=1        COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
  121=1        COMMON /CINOUT/ TDELAY,TIMER,BMOOP,PBMOOP
  122=1        COMMON /CINOUT/ SECFOF,PCFWNP,DLGFON
  123=1        COMMON /CINOUT/ OCTD,ORCLSE,OOTD,OROPEN,PCFWUP,DLGOFF,SLOWER
  124=1        COMMON /CINOUT/ SUDOWN,DLCVV,DHCVV,MNDPON,MDPON,TRKSTP,SURISE
  125=1        COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
  126=1        COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  127=1        COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
  128=1        COMMON /CINOUT/ PTVV,STVV
  129=1        COMMON /CALC/ APDCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  130=1        COMMON /CALC/ FDR,WJCNST
  131=1        COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  132=1        COMMON /CALC/ DPMSP,DSMSP,DDHMSP,PNT10,AUX9,AUX10
  133=1        COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
  134=1        COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
  135=1        COMMON /CALC/ PMTRQ,DPMP,R_PPT,PTREFF,REPST
  136=1        COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
  137=1        COMMON /CALC/ SMTRQ,DSMP,STREFF,REFT
  138=1        COMMON /CALC/ DPFRT,PDIFF,DPFWJ,DSFRT,SDIFF,DPSWJ
  139=1        COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
  140=1        COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
  141=1        COMMON /CALC/ SPDIS,SPDFF,MAXMSP,TREP,DES
  142=1        COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  143=1        COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  144=1        COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  145=1        COMMON /CALC/ DHMSPL,PAMPR,SAMPR,PPMPR,SPMPR
  146=1        COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
  147=1        COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
  148=1        COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
  149=1        COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
  150=1        COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
          C
          C    CONVERT ACTUAL ENGINE SPEEED
          C
  151          AENSP=FLOAT(IAENSP)*600.0/134.0
  152          IF(ABS(M1-AENSP).GT.300.0) THEN
  153            IF(AENSP .GT. M1) THEN
  154                    AENSP=M1+300.0
  155            ELSE
  156                    AENSP=M1-300.0
  157            ENDIF
  158          ENDIF
  159          M1=AENSP
          C
          C    CONVERT ACTUAL MOTOR AND ACTUAL SPROCKET SPEEDS
          C
  160          APMSP=FLOAT(IAPMSP)*600.0/24.0
  161          ASMSP=FLOAT(IASMSP)*600.0/24.0
  162          APSSP=FLOAT(IAPSSP)*600.0/304.0
          C    IF(ABS(APSSP-OAPSSP).GT.5.0) THEN
          C      IF(APSSP.GT.OAPSSP) THEN
```

```
        C                 APSSP=OAPSSP+5.0
        C             ELSE
        C                 APSSP=OAPSSP-5.0
        C             ENDIF
        C
        C         ENDIF
        C         ASSSP=FLOAT(IASSSP)*600.0/304.0
        C         IF(ABS(ASSSP-OASSSP).GT.5.0) THEN
        C             IF(ASSSP.GT.OASSSP) THEN
        C                 ASSSP=OASSSP+5.0
        C             ELSE
        C                 ASSSP=OASSSP-5.0
        C             ENDIF
        C         ENDIF
        C
        C         DPSP=ABS(OAPSSP-APSSP)*FDR
        C         DSSP=ABS(OASSSP-ASSSP)*FDR
        C         IF((DTRST .EQ. DRIVE).OR.(DTRST .EQ. REVRSE)) THEN
        C         IF(ABS(OAPMSP - APMSP).GT.DPSP) THEN
        C           IF(APMSP .GT. OAPMSP) THEN
        C                   APMSP=APMSP+DPSP
        C           ELSE
        C                   APMSP=APMSP-DPSP
        C           ENDIF
        C         ENDIF
        C         IF(ABS(OASMSP - ASMSP).GT.DSSP) THEN
        C           IF(ASMSP .GT. OSFMSP) THEN
        C                   ASMSP=ASMSP+DSSP
        C           ELSE
        C                   ASMSP=ASMSP-DSSP
        C           ENDIF
        C         ENDIF
        C         ELSE
        C           IF(ABS(OAPMSP-APMSP).GT.300.0) THEN
        C                   IF(APMSP .GT. OAPMSP) THEN
        C                           APMSP=APMSP+300.0
        C                   ELSE
        C                           APMSP=APMSP-300.0
        C                   ENDIF
        C           ENDIF
        C           IF(ADS(OASMSP -ASMSP) .GT. 300.0) THEN
        C                   IF(ASMSP .GT.OASMSP)THEN
        C                           ASMSP=ASMSP+300.0
        C                   ELSE
        C                           ASMSP=ASMSP-300.0
        C                   ENDIF
        C           ENDIF
        C         ENDIF
        C          MI=AENSP
        C          OAPSSP=APSSP
        C          OASSSP=ASSSP
        C          OAPMSP=APMSP
        C          OASMSP=ASMSP
        C         CONVERT WATERJET SPEEDS
        C
  163             APWSP=FLOAT(IAPWSP)*600.0/30.0
  164             ASWSP=FLOAT(IASWSP)*600.0/30.0
        C
```

```
C  CHECK FOR ENGINE MAG PICKUP FAILURE (DEFAULT TO 600 RPM)
C
          IF((AENSP.LT.400.0).AND.(LMCPSF).AND.(LSCPSF)) THEN
               AENSP=300.1
               AENSF=.TRUE.
          ENDIF
C
C         CHECK FOR MAG PICKUP FAILURES ON THE MOTORS, SPROCKETS
C          , AND WATERJETS IF IN DRIVE OR REVERSE :
C
          IF((DTRST.EQ.DRIVE).OR.(DTRST.EQ.REVRSE)) THEN
               IF((PRMOOP.EQ.LAND).OR.(PRMOOP.EQ.TRNSTN)) THEN
                    IF(ABS(APMSP-APSSP*FDR).GT.200) THEN
                         IF(DIM(APMSP,APSSP*FDR).EQ.0) THEN
                              APMSF=.TRUE.
                              APMSP=APSSP*FDR
                              ELSE
                              APSSF=.TRUE.
                              APSSP=APMSP/FDR
                         ENDIF
                    ENDIF
                    IF(ABS(ASMSP-ASSSP*FDR).GT.200) THEN
                         IF(DIM(ASMSP,ASSSP*FDR).EQ.0) THEN
                              ASMSF=.TRUE.
                              ASMSP=ASSSP*FDR
                              ELSE
                              ASSSF=.TRUE.
                              ASSSP=ASMSP/FDR
                         ENDIF
                    ENDIF
               ENDIF
               IF(PRMOOP.EQ.TRNSTN) THEN
                    IF((APWSP-APMSP*WJCNST).LT.-400) THEN
                         APWSF=.TRUE.
                         APWSP=APMSP*WJCNST
                    ENDIF
                    IF((ASWSP-ASMSP*WJCNST).LT.-400) THEN
                         ASWSF=.TRUE.
                         ASWSP=ASMSP*WJCNST
                    ENDIF
               ENDIF
               IF(PRMOOP.EQ.SEA) THEN
                    IF(ABS(APWSP-ASWSP).GT.400) THEN
                         IF(DIM(APWSP,ASWSP).EQ.0) THEN
                              APWSF=.TRUE.
                              APWSP=ASWSP
                              ELSE
                              ASWSF=.TRUE.
                              ASWSP=APWSP
                         ENDIF
                    ENDIF
               ENDIF
          ENDIF
C
          RETURN
          END
```

```
003C   1E                       PUSH    DS
003D   2E8E1E3A00               MOV     DS,CS:@@DATA$FRAME
0042   55                       PUSH    BP
0043   8BEC                     MOV     BP,SP
0045   83EC1400                 SUB     SP,14H
                                                      ; STATEMENT # 151
0049   2E8E060000               MOV     ES,CS:@CONST
004E   9B26DB06C400             FLD     ES:IAENSP
0054   9B2ED80E0A00             FMUL    CS:@CONST+0AH; 7
005A   9B2ED8360E00             FDIV    CS:@CONST+0EH; 7
0060   2E8E060400               MOV     ES,CS:@CONST+4H
0065   9B26D91E7400             FSTP    ES:AENSP; 7
006B   9B                       WAIT
006C   2E8E060600               MOV     ES,CS:@CONST+6H
0071   9B26D9060000             FLD     ES:M1
0077   2E8E060400               MOV     ES,CS:@CONST+4H
007C   9B26D9067400             FLD     ES:AENSP; 7
0082   9BDDD2                   FST     @TOS+2H
0085   9BD9C9                   FXCH    @TOS+1H
0088   9BDDD3                   FST     @TOS+3H
008B   9BDEE1                   FSUBRP                ; 6
008E   9BD9E1                   FABS                  ; 7
0091   9B2ED9061200             FLD     CS:@CONST+12H; 6
0097   9BD9C9                   FXCH    @TOS+1H
009A   9BDED9                   FCOMPP                ; 6
009D   50                       PUSH    AX            ; 1
009E   9BDDBEEAFF               FSTSW   [BP].@STACK+2H
00A3   58                       POP     AX            ; 1
00A4   9E                       SAHF
00A5   9BDB7EF6                 FSTP    [BP].@TEMP3+0AH
00A9   7703                     JA      $+5H
00AB   E94F00                   JMP     @@000000
00AE   9BDB6EF6                 FLD     [BP].@TEMP3+0AH
00B2   9BD9C1                   FLD     @TOS+1H
00B5   9BD9C9                   FXCH    @TOS+1H
00B8   9BDED9                   FCOMPP                ; 7
00BB   50                       PUSH    AX            ; 1
00BC   9BDDBEEAFF               FSTSW   [BP].@STACK+2H; 1
00C1   58                       POP     AX            ; 1
00C2   9E                       SAHF
00C3   9BDB7EEC                 FSTP    [BP].@TEMP3+14H; 1
00C7   7703                     JA      $+5H
00C9   E91400                   JMP     @@000001
00CC   9BDB6EEC                 FLD     [BP].@TEMP3+14H; 1
00D0   9B2ED8061200             FADD    CS:@CONST+12H; 1
00D6   9B26D91E7400             FSTP    ES:AENSP; 1
00DC   9B                       WAIT
00DD   E91000                   JMP     @@000002
                    @@000001:
                                                      ; STATEMENT # 156
00E0   2E8E060600               MOV     ES,CS:@CONST+6H
00E5   9B26D9060000             FLD     ES:M1
00EB   9B2ED8261200             FSUB    CS:@CONST+12H; 7
00F1   2E8E060400               MOV     ES,CS:@CONST+4H
00F6   9B26D91E7400             FSTP    ES:AENSP; 7
00FC   9B                       WAIT
                    @@000002:
```

```
FORTRAN-86 COMPILER         GENERATED CODE
:F2:FREQIN.FOR

                    @@000000:
                                                              ; STATEMENT # 159
 00FD   2E8E060400            MOV       ES,CS:@CONST+4H
 0102   9B26D9067400          FLD       ES:AENSP
 0108   2E8E060600            MOV       ES,CS:@CONST+6H
 010D   9B26D91E0000          FSTP      ES:M1   ; 7
 0113   9B                    WAIT
 0114   2E8E060000            MOV       ES,CS:@CONST
 0119   9B26DB066C00          FLD       ES:IAPMSP
 011F   9B2ED9060A00          FLD       CS:@CONST+0AH; 6
 0125   9BDEC9                FMULP              ; 6
 0128   9B2ED9061600          FLD       CS:@CONST+16H; 6
 012E   9BDEF9                FDIVP              ; 6
 0131   2E8E060400            MOV       ES,CS:@CONST+4H
 0136   9B26D91E9400          FSTP      ES:APMSP; 7
 013C   9B                    WAIT
 013D   2E8E060000            MOV       ES,CS:@CONST
 0142   9B26DB066E00          FLD       ES:IASMSP
 0148   9B2ED9060A00          FLD       CS:@CONST+0AH; 6
 014E   9BDEC9                FMULP              ; 6
 0151   9B2ED8361600          FDIV      CS:@CONST+16H; 7
 0157   2E8E060400            MOV       ES,CS:@CONST+4H
 015C   9B26D91E9800          FSTP      ES:ASMSP; 7
 0162   9B                    WAIT
 0163   2E8E060000            MOV       ES,CS:@CONST
 0168   9B26DB06D000          FLD       ES:IAPSSP
 016E   9B2ED9060A00          FLD       CS:@CONST+0AH; 6
 0174   9BDEC9                FMULP              ; 6
 0177   9B2ED8361A00          FDIV      CS:@CONST+1AH; 7
 017D   2E8E060400            MOV       ES,CS:@CONST+4H
 0182   9B26D91ED400          FSTP      ES:APSSP; 7
 0188   9B                    WAIT
 0189   2E8E060000            MOV       ES,CS:@CONST
 018E   9B26DB06D800          FLD       ES:IAPWSP
 0194   9B2ED9060A00          FLD       CS:@CONST+0AH; 6
 019A   9BDEC9                FMULP              ; 6
 019D   9B2ED9061E00          FLD       CS:@CONST+1EH; 6
 01A3   9BDEF9                FDIVP              ; 6
 01A6   9BD91E0000            FSTP      APWSP    ; 7
 01AB   9B                    WAIT
 01AC   9B26DB06DC00          FLD       ES:IASWSP
 01B2   9B2ED80E0A00          FMUL      CS:@CONST+0AH; 7
 01B8   9B2ED8361E00          FDIV      CS:@CONST+1EH; 7
 01BE   9BD91E0400            FSTP      ASWSP    ; 7
 01C3   9B                    WAIT
 01C4   9B2ED9062200          FLD       CS:@CONST+22H; 7
 01CA   2E8E060400            MOV       ES,CS:@CONST+4H
 01CF   9B26D81E7400          FCOMP     ES:AENSP; 7
 01D5   50                    PUSH      AX      ; 1
 01D6   9BDDBEEAFF            FSTSW     [BP].@STACK+2H
 01DB   58                    POP       AX      ; 1
 01DC   9E                    SAHF
 01DD   7703                  JA        $+5H
 01DF   E94000                JMP       @@000003
 01E2   2E8E060800            MOV       ES,CS:@CONST+8H
 01E7   26F606290001          TEST      ES:LPCPSF,1H
 01ED   7503                  JNZ       $+5H
```

```
01EF  E93000              JMP      @@000003
01F2  26F6062C0001        TEST     ES:LSCP3F,1H
01F8  7503                JNZ      $+5H
01FA  E92500              JMP      @@000003
01FD  9B2ED9062600        FLD      CS:@CONST+26H; 7
0203  2E8E060400          MOV      ES,CS:@CONST+4H
0208  9B26D91E7400        FSTP     ES:AENSP; 7
020E  9B                  WAIT
020F  2E8E060800          MOV      ES,CS:@CONST+8H
0214  26C70610010100      MOV      ES:AENSF,1H
021B  26C70612010000      MOV      ES:AENSF+2H,0H
                @@000003:
                                                    ; STATEMENT # 169
0222  2E8E060000          MOV      ES,CS:@CONST
0227  26A1062000          MOV      AX,ES:DTRST
022C  268B16E200          MOV      DX,ES:DTRST+2H
0231  268B0E8000          MOV      CX,ES:DRIVE
0236  268B1E8200          MOV      BX,ES:DRIVE+2H
023B  87D9                XCHG     BX,CX
023D  50                  PUSH     AX         ; 1
023E  52                  PUSH     DX         ; 2
023F  9A00000000          CALL     TQ_150
0244  58                  POP      AX         ; 2
0245  5A                  POP      DX         ; 1
0247  7503                JNZ      $+5H
0249  E92000              JMP      @@000005
024C  2E8E060000          MOV      ES,CS:@CONST
0250  268B0E7C00          MOV      CX,ES:REVRSE
0255  268B1E7E00          MOV      BX,ES:REVRSE+2H
025A  87C2                XCHG     AX,DX
025C  87D9                XCHG     BX,CX
025E  9A00000000          CALL     TQ_150
0263  7503                JNZ      $+5H
0265  E90300              JMP      @@000005
0268  E95D03              JMP      @@000004
                @@000005:
                                                    ; STATEMENT # 170
026B  2E8E060000          MOV      ES,CS:@CONST
0270  26A1062400          MOV      AX,ES:PRMOOP
0275  268B16E600          MOV      DX,ES:PRMOOP+2H
027A  268B0E5800          MOV      CX,ES:LAND
027F  268B1E6A00          MOV      BX,ES:LAND+2H
0284  87D9                XCHG     BX,CX
0286  50                  PUSH     AX         ; 1
0287  52                  PUSH     DX         ; 2
0288  9A00000000          CALL     TQ_150
028D  58                  POP      AX         ; 2
028E  5A                  POP      DX         ; 1
028F  7503                JNZ      $+5H
0291  E92300              JMP      @@000007
0294  2E8E060000          MOV      ES,CS:@CONST
0299  268B0E6E00          MOV      CX,ES:TRNSTN
029E  268B1E6E00          MOV      BX,ES:TRNSTN+2H
02A3  87C2                XCHG     AX,DX
02A5  87D9                XCHG     BX,CX
02A7  9A00000000          CALL     TQ_150
02AC  7503                JNZ      $+5H
```

```
02AE   E90300              JMP     @@000007
~~02B1   E98A01              JMP     @@000006~~
~~               @@000007:~~
~~                                               ; STATEMENT # 171~~
~~02B4   2E8E060400          MOV     ES,CS:@CONST+4H~~
~~02B9   9B26D906B400        FLD     ES:APSSP~~
~~02BF   9B26D80E1000        FMUL    ES:FDR   ; 7~~
~~02C5   9B26D9069400        FLD     ES:APMSP; 7~~
~~02CB   9BDDD2              FST     @TOS+2H~~
~~02CE   9BD9C9              FXCH    @TOS+1H~~
~~02D1   9BDDD3              FST     @TOS+3H~~
~~02D4   9BDEE9              FSUBP            ; 6~~
~~02D7   9BD9E1              FABS             ; 7~~
~~02DA   9B2EDF062A00        FLD     ES:@CONST+2AH; 7~~
~~02E0   9BD9C9              FXCH    @TOS+1H~~
~~02E3   9BDED9              FCOMPP           ; 6~~
~~02E6   50                  PUSH    AX       ; 1~~
~~02E7   9BDDBEEAFF          FSTSW   [BP].@STACK+2H~~
~~02EC   58                  POP     AX       ; 1~~
~~02ED   9E                  SAHF~~
~~02EE   9BDD7EF6            FSTP    [BP].@TEMPS+0AH~~
~~02F2   7703                JA      $+5H~~
~~02F4   E96200              JMP     @@000008~~
~~02F7   9BDD6EF6            FLD     [BP].@TEMPS+0AH~~
~~02FB   9BD9C9              FXCH    @TOS+1H~~
~~02FE   9A00000000          CALL    MQERDIM~~
~~0303   9B2EDF062E00        FLD     ES:@CONST+2EH; 1~~
~~0309   9BDED9              FCOMPP~~
~~030C   8BC0                MOV     AX,AX~~
~~030E   50                  PUSH    AX       ; 1~~
~~030F   9BDDBEEAFF          FSTSW   [BP].@STACK+2H; 2~~
~~0314   58                  POP     AX       ; 1~~
~~0315   250041              AND     AX,4100H~~
~~0318   350040              XOR     AX,4000H~~
~~031B   7403                JE      $+5H~~
~~031D   E92E00              JMP     @@000009~~
~~0320   2E8E0608            MOV     ES,CS:@CONST+8H~~
~~0325   26C7061800.0100     MOV     ES:APMSP,1H~~
~~032C   26C7061A010000      MOV     ES:APMSP+2H,0H~~
~~0333   2E8E060400          MOV     ES,CS:@CONST+4H~~
~~0338   9B26D906B400        FLD     ES:APSSP; 2~~
~~033E   9B26D80E1000        FMUL    ES:FDR   ; 1~~
~~0344   9B26D91E9400        FSTP    ES:APMSP; 1~~
~~034A   9B                  WAIT~~
~~034B   E92B00              JMP     @@000010~~
~~               @@000009:~~
~~                                               ; STATEMENT # 176~~
~~034E   2E8E060800          MOV     ES,CS:@CONST+8H~~
~~0353   26C70614010100      MOV     ES:APSSP,1H~~
~~035A   26C70616010000      MOV     ES:APSSP+2H,0H~~
~~0361   2E8E060400          MOV     ES,CS:@CONST+4H~~
~~0366   9B26D9069400        FLD     ES:APMSP~~
~~036C   9B26D8361000        FDIV    ES:FDR   ; 7~~
~~0372   9B26D91EB400        FSTP    ES:APSSP; 7~~
~~0378   9B                  WAIT~~
~~               @@000010:~~
~~               @@000008:~~
```

```
                                                    ; STATEMENT # 180
   0379   2E8E060400         MOV      ES,CS:@CONST+4H
   037E   9B26D906D800       FLD      ES:ASSSP
   0384   9B26D80E1800       FMUL     ES:FDR   ; 7
   038A   9B26D9069800       FLD      ES:ASMSP; 7
   0390   9BDDD2             FST      @TOS+2H
   0393   9BD9C9             FXCH     @TOS+1H
   0396   9BDDD3             FST      @TOS+3H
   0399   9BDEE9             FSUBP                ; 6
   039C   9BD9E1             FABS                 ; 7
   039F   9B2EDF062A00       FLD      CS:@CONST+2AH; 7
   03A5   9BD9C9             FXCH     @TOS+1H
   03A8   9BDED9             FCOMPP               ; 6
   03AB   50                 PUSH     AX          ; 1
   03AC   9BDDBEEAFF         FSTSW    [BP].@STACK+2H
   03B1   58                 POP      AX          ; 1
   03B2   9E                 SAHF
   03B3   9BDB7EF6           FSTP     [BP].@TEMP3+0AH
   03B7   7703               JA       $+5H
   03B9   E98200             JMP      @@000011
   03BC   9BDB6EF6           FLD      [BP].@TEMP3+0AH
   03C0   9BD9C9             FXCH     @TOS+1H
   03C3   9A00000000         CALL     MQERDIM
   03C8   9B2EDF062E00       FLD      CS:@CONST+2EH; 1
   03CE   9BDED9             FCOMPP
   03D1   3BC0               MOV      AX,AX
   03D3   50                 PUSH     AX          ; 1
   03D4   9BDDBEEAFF         FSTSW    [BP].@STACK+2H; 2
   03D9   58                 POP      AX          ; 1
   03DA   250041             AND      AX,4100H
   03DD   350040             XOR      AX,4000H
   03E0   7403               JE       $+5H
   03E2   E92E00             JMP      @@000012
   03E5   2E8E060800         MOV      ES,CS:@CONST+8H
   03EA   26C70620010100     MOV      ES:ASMSP,1H
   03F1   26C70622010000     MOV      ES:ASMSP+2H,0H
   03F8   2E8E060400         MOV      ES,CS:@CONST+4H
   03FD   9B26D906D800       FLD      ES:ASSSP; 2
   0403   9B26D80E1800       FMUL     ES:FDR   ; 1
   0409   9B26D91E9800       FSTP     ES:ASMSP; 1
   040F   9B                 WAIT
   0410   E92800             JMP      @@000013
                       @@000012:
                                                    ; STATEMENT # 185
   0413   2E8E060800         MOV      ES,CS:@CONST+8H
   0418   26C7061C010100     MOV      ES:ASSSP,1H
   041F   26C7061E010000     MOV      ES:ASSSP+2H,0H
   0426   2E8E060400         MOV      ES,CS:@CONST+4H
   042B   9B26D9069800       FLD      ES:ASMSP
   0431   9B26D8361800       FDIV     ES:FDR   ; 7
   0437   9B26D91ED800       FSTP     ES:ASSSP; 7
   043D   9B                 WAIT
                       @@000013:
                       @@000011:
                       @@000006:
                                                    ; STATEMENT # 190
   043E   2E8E060000         MOV      ES,CS:@CONST
```

```
0443   268B06E400          MOV      AX,ES:PRMOOP
0448   268B16E600          MOV      DX,ES:PRMOOP+2H
044D   268B0E6800          MOV      CX,ES:TRNSTN
0452   268B1E6E00          MOV      BX,ES:TRNSTN+2H
0457   87D9                XCHG     BX,CX
0459   9A00000000          CALL     TQ_150
045E   7403                JZ       $+5H
0460   E9AC00              JMP      @@000014
0463   2E8E060400          MOV      ES,CS:@CONST+4H
0468   9B26D9069400        FLD      ES:APWSP
046E   9B26D80E1C00        FMUL     ES:WJCNST; 7
0474   9BD82E0000          FSUBR    APWSP      ; 7
0479   9B2EDF063200        FLD      CS:@CONST+32H; 7
047F   9BDED9              FCOMPP              ; 6
0482   50                  PUSH     AX         ; 1
0483   9BDDBEEAFF          FSTSW    [BP].@STACK+2H
0488   58                  POP      AX         ; 1
0489   9E                  SAHF
048A   7703                JA       $+5H
048C   E92A00              JMP      @@000015
048F   2E8E060800          MOV      ES,CS:@CONST+8H
0494   26C70624010100      MOV      ES:APWSF,1H
049B   26C70626010000      MOV      ES:APWSF+2H,0H
04A2   2E8E060400          MOV      ES,CS:@CONST+4H
04A7   9B26D9069400        FLD      ES:APWSP
04AD   9B26D80E1C00        FMUL     ES:WJCNST; 7
04B3   9BD91E0000          FSTP     APWSP      ; 7
04B8   9B                  WAIT
                  @@000015:
                                                   ; STATEMENT # 195
04B9   2E8E060400          MOV      ES,CS:@CONST+4H
04BE   9B26D9069800        FLD      ES:ASWSP
04C4   9B26D80E1C00        FMUL     ES:WJCNST; 7
04CA   9BD82E0400          FSUBR    ASWSP      ; 7
04CF   9B2EDF063200        FLD      CS:@CONST+32H; 7
04D5   9BDED9              FCOMPP              ; 6
04D8   50                  PUSH     AX         ; 1
04D9   9BDDBEEAFF          FSTSW    [BP].@STACK+2H
04DE   58                  POP      AX         ; 1
04DF   9E                  SAHF
04E0   7703                JA       $+5H
04E2   E92A00              JMP      @@000016
04E5   2E8E060800          MOV      ES,CS:@CONST+8H
04EA   26C70628010100      MOV      ES:ASWSF,1H
04F1   26C7062A010000      MOV      ES:ASWSF+2H,0H
04F8   2E8E060400          MOV      ES,CS:@CONST+4H
04FD   9B26D9069800        FLD      ES:ASWSP
0503   9B26D80E1C00        FMUL     ES:WJCNST; 7
0509   9BD91E0400          FSTP     ASWSP      ; 7
050E   9B                  WAIT
                  @@000016:
                  @@000014:
                                                   ; STATEMENT # 200
050F   2E8E060000          MOV      ES,CS:@CONST
0514   268B06E400          MOV      AX,ES:PRMOOP
0519   268B16E600          MOV      DX,ES:PRMOOP+2H
051E   268B0E7000          MOV      CX,ES:SEA
```

```
FORTRAN-86 COMPILER        GENERATED CODE
:F2:FREQIN.FOR

0523   268B1E7200          MOV      BX,ES:SEA+2H
0528   87D9                XCHG     BX,CX
052A   9A00000000          CALL     TQ_150
052F   7403                JZ       $+5H
0531   E99400              JMP      @@000017
0534   9BD9060000          FLD      APWSP     ; 7
0539   9BD9060400          FLD      ASWSP     ; 6
053E   9BDEE9              FSUBP              ; 6
0541   9BD9E1              FABS               ; 7
0544   9B2EDF063600        FLD      CS:@CONST+36H; 7
054A   9BD9C9              FXCH     @TOS+1H
054D   9BDED9              FCOMPP             ; 6
0550   50                  PUSH     AX        ; 1
0551   9BDDBEEAFF          FSTSW    [BP].@STACK+2H
0556   58                  POP      AX        ; 1
0557   9E                  SAHF
0558   7703                JA       $+5H
055A   E96B00              JMP      @@000018
055D   9BD9060000          FLD      APWSP     ; 7
0562   9BD9060400          FLD      ASWSP     ; 6
0567   9A00000000          CALL     MQERDIM
056C   9B2EDF062E00        FLD      CS:@CONST+2EH; 7
0572   9BDED9              FCOMPP             ; 6
0575   8BC0                MOV      AX,AX
0577   50                  PUSH     AX        ; 1
0578   9BDDBEEAFF          FSTSW    [BP].@STACK+2H
057D   58                  POP      AX        ; 1
057E   250041              AND      AX,4100H
0581   350040              XOR      AX,4000H
0584   7403                JE       $+5H
0586   E92100              JMP      @@000019
0589   2E8E060800          MOV      ES,CS:@CONST+8H
058E   26C70624010100      MOV      ES:APWSP,1H
0595   26C70626010000      MOV      ES:APWSP+2H,0H
059C   9BD9060400          FLD      ASWSP     ; 7
05A1   9BD91E0000          FSTP     APWSP     ; 7
05A6   9B                  WAIT
05A7   E91E00              JMP      @@000020
                     @@000019:
                                                  ; STATEMENT # 206
05AA   2E8E060800          MOV      ES,CS:@CONST+8H
05AF   26C70628010100      MOV      ES:ASWSP,1H
05B6   26C7062A010000      MOV      ES:ASWSP+2H,0H
05BD   9BD9060000          FLD      APWSP     ; 7
05C2   9BD91E0400          FSTP     ASWSP     ; 7
05C7   9B                  WAIT
                     @@000020:
                     @@000018:
                     @@000017:
                     @@000004:
                                                  ; STATEMENT # 212
05C8   E90000              JMP      @@000021
                     @@000021:
                                                  ; STATEMENT # 213
05CB   8BE5                MOV      SP,BP
05CD   5D                  POP      BP
05CE   1F                  POP      DS
```

```
05CF   CB                    RET
```

```
FORTRAN-86 COMPILER        ERROR MESSAGES AND SUMMARY
:F2:FREQIN.FOR

STORAGE REQUIREMENTS FOR MODULE FREQIN:

      CODE AREA SIZE          00596H      1430D
      CONSTANT AREA SIZE      0003AH        58D
      VARIABLE AREA SIZE      0000CH        12D
      MAXIMUM STACK SIZE      00020H        32D
      /ERROR/                 0012CH       300D
      /MOUT/                  000C6H       198D
      /CINOUT/                0019CH       412D
      /BITFNC/                00268H       616D
      /CALC/                  0020CH       524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 3CH
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF FREQIN COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY: :F1:FORT86.86  :F2:TSTREP.FOR

     1              SUBROUTINE TSTREP
    =1 $            INCLUDE (:F2:COMMON.FOR)
     2=1            INTEGER*4 RESULT,MASK(16),NAME
     3=1            INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
     4=1            INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
     5=1            INTRGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
     6=1            INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
     7=1            INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
      1             INTEGER*4 BIT19,BIT29,BIT39,BIT49
     9 1            INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
    10=1            INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
    11=1            INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
    12=1            INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
    13=1            INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
    14=1            INTEGER*4 JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
    15=1            INTEGER*4 BG1,BG2,BG3,BG4,BG5,BG6,BG7,BG8,BG9,BG10
    16=1            INTEGER*4 BG11,BG12,BG13,BG14,BG15,BG16,BG17,BG18,BG19,BG20
    17=1            INTEGER*4 BG21,BG22,BG23,BG24,BG25,NORMAL,POWER
    18=1            INTEGER*4 BG26,BG27,BG28,BG29,BG30
    19=1            INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
    20=1            LOGICAL*4 MGSF,TGSF,GSSF,BKNPSF,PGSCSF,INBPSF
    21=1            LOGICAL*4 HEOTSF,LBOLSF,LBRPSF,LBSPSF,LPCPSF
    22=1            LOGICAL*4 LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
    23=1            LOGICAL*4 TBIT,LGPPSF,FIRESF
    24=1            LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
    25=1            LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
    26=1            LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
    27=1            LOGICAL*4 AESPSF,SEWPSF,HSPVSF,FESPSF,PSCSSF
    28=1            LOGICAL*4 RELJSF,LSTRNS,SLTRNS,INTIAL,SFTINP
    29=1            LOGICAL*4 APBSSF,ASBCSF,BHMHSF,BHMLSF,BHMRSF,BTRNSF
    30=1            LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
    31=1            LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
    32=1            LOGICAL*4 MOOPGG,MLTSSF,RENTRY
    33=1            LOGICAL*4 NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
    34=1            LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
    35=1            INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
    36=1            INTEGER*4 IAPBCN,IASBCN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
    37=1            INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IBTRNR
    38=1            INTEGER*4 TRNDIR
    39=1            INTEGER*4 DIGIN(8),SUSMSG,NSUSMG
    40=1            INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
    41=1            INTEGER*4 TEST1,TEST2,TEST3,HYSPSS,HYNSPS,ENSNGW,ENSMSG
    42=1            INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
    43=1            INTEGER*4 CKPSLT,CKRSL1,CKRSL2
    44=1            INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
    45=1            INTEGER*4 DTRST,PRMOOP
    46=1            INTEGER*4 TDELAY,TIMER,DMOOP,PDMOOP,DIGOUT(8)
    47=1            INTEGER*4 SEGFOF,PGFWNP,BLGPON
    48=1            INTEGER*4 SSTB,SRGLSE,SOTB,SROPEN,PGFWBP,BLGOFF,SLOWER
    49=1            INTEGER*4 GUBOWN,BLGVV,BHGVV,MNBPON,MBPON,TRPSTP,SURISE
    50=1            INTEGER*4 SAFELY,BTBGS,PBTBGS,PTBGS
    51=1            INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IBEG
    52=1            INTEGER*4 RSMG,RPMG,RSGS,RPSS,RES
    53=1            INTEGER*4 PFVV,GTVV
    54=1            REAL*4 APBCAN,ASBCAN,BHMSP,BHMSR,BTRNR,PAMTPR
```

```
   55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
   56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
   57=1          REAL*4 DPMSP,DSMSP,DDHMSP,PNT16,AUX9,AUX10
   58=1          REAL*4 DHWSP,DHWSK,DPWSP,DSWSP
   59=1          REAL*4 APPS,ASPS,AENSP,PMDIS,PMDFF
   60=1          REAL*4 PMTRQ,DPMP,REPPT,PTREFF,REPST
   61=1          REAL*4 APMSP,ASMSP,SMDIS,SMDFF
   62=1          REAL*4 SMTRQ,DSMP,STREFF,REPT
   63=1          REAL*4 DPPRT,PDIFF,DPPWJ,DSPRT,SDIFF,DPSWJ
   64=1          REAL*4 PPDIS,PPDFF,APSSP,ASSSP
   65=1          REAL*4 PPTRQ,DPPP,REPPP,PPMEFF,REPSP
   66=1          REAL*4 SPDIS,SPDFF,MAXMSP,TREP,DES
   67=1          REAL*4 SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
   68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
   69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
   70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
   71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
   72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
   73=1          REAL*4 M29,M30,M31,M32,M33,M34
   74=1          REAL*4 DHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
   75=1          INTEGER*4 M2,M3,M10,M11,M12
   76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
   77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
   78=1          COMMON /BITFNC/ RESULT,MASK,NAME
   79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
   81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
   82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
   83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
   84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
   85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
   86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
   87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
   88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
   89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
   90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
   91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
   92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
   93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
   94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
   95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
   96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
   97=1          COMMON /ERROR/MSSF,TSSF,DSSF,LKNPSF,POSCSF,INDPSF
   98=1          COMMON /ERROR/ HEOTSF,LDCLSF,LDKPSF,LSSPSF,LPCPSF
   99=1          COMMON /ERROR/ LSOPSF,HFDPSF,LEOPSF,LPHLSF,LSHLSF
  100=1          COMMON /ERROR/ LSPPSF,FIRESF
  101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
  104=1          COMMON /ERROR/ AEBPSF,SEWPSF,HBPVSF,FEBPSF,POCSSF
  105=1          COMMON /ERROR/ RCLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
  106=1          COMMON /ERROR/ APSCSF,ASSCSF,DHMHSF,DHMLSF,DMMRSF,DTRNSF
  107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  109=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
  110=1          COMMON /ERROR/ NOFAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
  111=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  112=1          COMMON /CINOUT/ INPDCN,INSDCN,IDHMSP,IDHMSK,IPAMPR,IPFMPR
  113=1          COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  114=1          COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
  115=1          COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  116=1          COMMON /CINOUT/ TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNGN,ENGMSG
  117=1          COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
  118=1          COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
  119=1          COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  120=1          COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDBS,PDTDBS,PTDBS
  121=1          COMMON /CINOUT/ TDELAY,TIMER,BMOOP,PBMOOP
  122=1          COMMON /CINOUT/ GECFGF,PCFWNP,DLGPON
  123=1          COMMON /CINOUT/ GCTD,GRCLSE,GOTR,GROPEN,PCFWSP,DLGOFF,SLOWER
  124=1          COMMON /CINOUT/ GUDOWN,DLGVV,DHGVV,MNDPON,MDPON,TRKSTP,SURISE
  125=1          COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
  126=1          COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
  127=1          COMMON /CINOUT/ RGMS,RPMS,RGSS,RPSS,RES
  128=1          COMMON /CINOUT/ PTVV,STVV
  129=1          COMMON /CALC/ APDCAN,ASDCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  130=1          COMMON /CALC/ FDR,WJCNST
  131=1          COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
  132=1          COMMON /CALC/ DPMSP,DSMSP,DDHMSP,PNT18,AUX9,AUX10
  133=1          COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
  134=1          COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDPP
  135=1          COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
  136=1          COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDPP
  137=1          COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
  138=1          COMMON /CALC/ DPFRT,PDIFP,DPPWJ,DSFRT,SDIFP,DPSWJ
  139=1          COMMON /CALC/ PPDIS,PPDPP,APSSP,ASSSP
  140=1          COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFP,REPSP
  141=1          COMMON /CALC/ SPDIS,SPDPP,MAXMSP,TREP,DES
  142=1          COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  143=1          COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  144=1          COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  145=1          COMMON /CALC/ DHMSPD,PAMPR,SAMPR,PFMPR,SFMPR
  146=1          COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
  147=1          COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
  148=1          COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
  149=1          COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
  150=1          COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
  151            MINUS=-1.0
           C
           C     CALCULATE PORT WATERJET MOTREP
           C
  152            DPFRT=DPWSP*6.43
  153            PDIFP=PFMTPR-PAMTPR
  154            IF((DTRST.EQ.REVRSE).AND.(PMOOP.EQ.TRNSTN))
               +      PDIFP=PDIFP*MINUS
  155            IF((PAMHSF).OR.(PAMLSF).OR.(PFMHSF).OR.(PFMLSF)) PDIFP=2000.0
  156            DPPWJ=DPFRT*PDIFP/395934.0
  157            PTREFF=(ABS(DPPWJ/2497))**0.25
  158            REPPT=DPPWJ/PTREFF
           C
           C     CALCULATE STARBOARD WATERJET MOTREP
           C
  159            DSFRT=DSWSP*6.43
  160            SDIFP=SFMTPR-SAMTPR
  161            IF((DTRST.EQ.REVRSE).AND.(PMOOP.EQ.TRNSTN))
               +      SDIFP=SDIFP*MINUS
  162            IF((SAMHSF).OR.(SAMLSF).OR.(SFMHSF).OR.(SFMLSF)) SDIFP=2000.0
```

```
 163          DPSWJ=DSFRT*SDIFP/395934.0
 164          STREFF=(ABS(DPSWJ/249))**0.25
 165          REPST=DPSWJ/STREFF
       C
       C      REQUIRED POWER FOR TRANSMISSIONS
       C
 166          REPT=REPPT+REPST
       C
 167          RETURN
 168          END
```

```
FORTRAN-86 COMPILER        ERROR MESSAGES AND SUMMARY
:F2:TSTREP.FOR

STORAGE REQUIREMENTS FOR MODULE TSTREP:

    CODE AREA SIZE       002B8H      696D
    CONSTANT AREA SIZE   0001EH       30D
    VARIABLE AREA SIZE   0000AH       10D
    MAXIMUM STACK SIZE   00000H        0D
    /ERROR/              0012CH      300D
    /MOUT/               000C6H      198D
    /SINOUT/             0019CH      412D
    /BITFNG/             00268H      616D
    /SAL8/               0020CH      524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 20H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF TSTREP COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

```
FORTRAN-86 COMPILER
:F2:UPSHFT.FOR

SERIES-III FORTRAN-86 COMPILER V1.0
COMPILER INVOKED BY: :F1:FORT86.86  :F2:UPSHFT.FOR

   1           SUBROUTINE UPSHFT
    =1 $       INCLUDE (:F2:COMMON.FOR)
   2=1         INTEGER*4 RESULT,MASK(16),NAME
   3=1         INTEGER*4 BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
   4=1         INTEGER*4 BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17,BIT18
   5=1         INTEGER*4 BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27,BIT28
   6=1         INTEGER*4 BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37,BIT38
   7=1         INTEGER*4 BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47,BIT48
   8=1         INTEGER*4 BIT19,BIT29,BIT39,BIT49
   9=1         INTEGER*4 BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7,BEGIN8
  10=1         INTEGER*4 BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15,BEGN16
  11=1         INTEGER*4 WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7,WIDTH8
  12=1         INTEGER*4 WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15,WIDT16
  13=1         INTEGER*4 JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7,JCARD8
  14=1         INTEGER*4 JCAR9,JCAR10,JCAR11, JCAR12,JCAR13,JCAR14,JCAR15,JCAR16
  15=1         INTEGER*4 DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
  16=1         INTEGER*4 DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
  17=1         INTEGER*4 DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
  18=1         INTEGER*4 DG26,DG27,DG28,DG29,DG30
  19=1         INTEGER*4 WIDTH,BEGIN,END,RMASK,I,K
  20=1         LOGICAL*4 MSSF,TSSF,OSSF,DKNPSF,FGSCSF,INDPSF
  21=1         LOGICAL*4 HEOTSF,LBCLSF,LJRPSF,LBSPSF,LPCPSF
  22=1         LOGICAL*4 LSCPSF,HFBPSF,LEOPSF,LPHLSF,LSHLSF
  23=1         LOGICAL*4 TBIT,LSPPSF,FIRESF
  24=1         LOGICAL*4 P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
  25=1         LOGICAL*4 S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
  26=1         LOGICAL*4 LVSVSF,LCSVSF,HVSVSF,HCSVSF
  27=1         LOGICAL*4 AESPSF,SEWPSF,HBPVSF,FEBPSF,FGCSSF
  28=1         LOGICAL*4 RCLSSF,LSTRNS,SLTRNS,INITIAL,SFTINP
  29=1         LOGICAL*4 APSCSF,ASSCSF,DHMHSF,DHMLSF,DHMRSF,DTRNSF
  30=1         LOGICAL*4 PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHSF,SFMLSF
  31=1         LOGICAL*4 SAMHSF,SAMLSF,ENCTSF,PHOTSF,SHOTSF
  32=1         LOGICAL*4 MOOPCC,MLTSSF,RENTKY
  33=1         LOGICAL*4 NOFAN,F4GPM,F8GPM,F12GPM,FTIME,NCLTCH
  34=1         LOGICAL*4 AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
  35=1         INTEGER*4 PATVV,PFTVV,SATVV,SFTVV
  36=1         INTEGER*4 IAPBCN,IASBCN,IDHMSP,IDHMSR,IPAMPR,  IPFMPR
  37=1         INTEGER*4 ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
  38=1         INTEGER*4 TRNDIR
  39=1         INTEGER*4 DIGIN(3),SUSMSG,NSUSMG
  40=1         INTEGER*4 LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
  41=1         INTEGER*4 TEST1,TEST2,TEST3,HYDPSS,HYNDPS,ENGNON,ENGMSG
  42=1         INTEGER*4 CKPUMP,CKRAMP,CKPLST,CKRLST
  43=1         INTEGER*4 CKPSLT,CKRSL1,CKRSL2
  44=1         INTEGER*4 IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
  45=1         INTEGER*4 DTRST,PRMOOP
  46=1         INTEGER*4 TDELAY,TIMER,DMOOP,PDMOOP,DIGOUT(8)
  47=1         INTEGER*4 SECPOP,PCPWNP,BLOPON
  48=1         INTEGER*4 OCTO,SRCLSE,GOTO,DROPEN,PCPWDP,BLOOFF,SLOWER
  49=1         INTEGER*4 SUBDWN,DLCVV,DHCVV,MNDPON,MSPON,TRNSTP,SURISE
  50=1         INTEGER*4 ERRDLY,DTDGS,PDTDGS,PTDGS
  51=1         INTEGER*4 AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,TDGS
  52=1         INTEGER*4 RSMS,RPMS,RSSS,RPSS,RES
  53=1         INTEGER*4 PTVV,STVV
  54=1         REAL*4 APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
  55=1          REAL*4 FDR,WJCNST,INTCPT(-1:1)
  56=1          REAL*4 PFMTPR,SFMTPR,SAMTPR,ENGTEM,PHOTEM,SHOTEM
  57=1          REAL*4 DPMSP,DSMSP,DBHMSP,PNT1S,AUX9,AUX10
  58=1          REAL*4 BHWSP,DHWSR,DPWSP,DSWSP
  59=1          REAL*4 APPS,ASPS,AENSP,PMBIS,PMBFP
  60=1          REAL*4 PMTRQ,BPMP,REPPY,PTREFF,REPST
  61=1          REAL*4 APMSP,ASMSP,SMBIS,SMBFP
  62=1          REAL*4 SMTRQ,BSMP,STREFF,REPT
  63=1          REAL*4 DPFRT,PBIFP,DPPWJ,DSFRT,SBIFP,BPSWJ
  64=1          REAL*4 PPBIS,PPBFP,APSSP,ASSSP
  65=1          REAL*4 PPTRQ,BPPP,REPPP,PPMEFF,REPSP
  66=1          REAL*4 SPBIS,SPBFP,MAXMSP,TREP,BES
  67=1          REAL*4 SPTRQ,BSPP,SPMEFF,REPP,ALPMSP,ALSMSP
  68=1          REAL*4 ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
  69=1          REAL*4 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
  70=1          REAL*4 M1,M4,M5,M6,M7,M8,M9
  71=1          REAL*4 M13,M14,M15,M16,M17,M18,M19,M20
  72=1          REAL*4 M21,M22,M23,M24,M25,M26,M27,M28
  73=1          REAL*4 M29,M30,M31,M32,M33,M34
  74=1          REAL*4 BHMSPB(4),PAMPR(10),SAMPR(10),PFMPR(10),SFMPR(10)
  75=1          INTEGER*4 M2,M3,M10,M11,M12
  76=1          INTEGER*4 M35,M36,M37,M38,M39,M40
  77=1          INTEGER*4 M41,M42,M43,M44,M45,M46,M47,M48,M49
  78=1          COMMON /BITFNC/ RESULT,MASK,NAME
  79=1          COMMON /BITFNC/ BIT,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7,BIT8,BIT9
  80=1          COMMON /BITFNC/ BIT10,BIT11,BIT12,BIT13,BIT14,BIT15,BIT16,BIT17
  81=1          COMMON /BITFNC/ BIT20,BIT21,BIT22,BIT23,BIT24,BIT25,BIT26,BIT27
  82=1          COMMON /BITFNC/ BIT30,BIT31,BIT32,BIT33,BIT34,BIT35,BIT36,BIT37
  83=1          COMMON /BITFNC/ BIT40,BIT41,BIT42,BIT43,BIT44,BIT45,BIT46,BIT47
  84=1          COMMON /BITFNC/ BIT19,BIT29,BIT39,BIT49,BIT18,BIT28,BIT38,BIT48
  85=1          COMMON /BITFNC/ BEGIN1,BEGIN2,BEGIN3,BEGIN4,BEGIN5,BEGIN6,BEGIN7
  86=1          COMMON /BITFNC/ BEGN9,BEGN10,BEGN11,BEGN12,BEGN13,BEGN14,BEGN15
  87=1          COMMON /BITFNC/ WIDTH1,WIDTH2,WIDTH3,WIDTH4,WIDTH5,WIDTH6,WIDTH7
  88=1          COMMON /BITFNC/ WIDT9,WIDT10,WIDT11,WIDT12,WIDT13,WIDT14,WIDT15
  89=1          COMMON /BITFNC/ BEGIN8,BEGN16,WIDTH8,WIDT16,JCARD8,JCAR16
  90=1          COMMON /BITFNC/ JCARD1,JCARD2,JCARD3,JCARD4,JCARD5,JCARD6,JCARD7
  91=1          COMMON /BITFNC/ JCAR9,JCAR10,JCAR11,JCAR12,JCAR13,JCAR14,JCAR15
  92=1          COMMON /BITFNC/ DG1,DG2,DG3,DG4,DG5,DG6,DG7,DG8,DG9,DG10
  93=1          COMMON /BITFNC/ DG11,DG12,DG13,DG14,DG15,DG16,DG17,DG18,DG19,DG20
  94=1          COMMON /BITFNC/ DG21,DG22,DG23,DG24,DG25,NORMAL,POWER
  95=1          COMMON /BITFNC/ DG26,DG27,DG28,DG29,DG30
  96=1          COMMON /BITFNC/ WIDTH,BEGIN,END,RMASK,I,K
  97=1          COMMON /ERROR/MSSF,TSSF,GSSF,BKNPSF,PGSCSF,INDPSF
  98=1          COMMON /ERROR/ HEOTSF,LBCLSF,LBRPSF,LBSPSF,LPCPSF
  99=1          COMMON /ERROR/ LSCPSF,HPDPSF,LEOPSF,LPMLSF,LSMLSF
 100=1          COMMON /ERROR/ LSPPSF,FIRESF
 101=1          COMMON /ERROR/ P1SCSF,P2SCSF,P3SCSF,P4SCSF,P5SCSF
 102=1          COMMON /ERROR/ S1SCSF,S2SCSF,S3SCSF,S4SCSF,S5SCSF
 103=1          COMMON /ERROR/ LVSVSF,LCSVSF,HVSVSF,HCSVSF
 104=1          COMMON /ERROR/ ALBPSF,SEWPSF,HBPVSF,FEBPSF,PGBSSF
 105=1          COMMON /ERROR/ RSLSSF,LSTRNS,SLTRNS,INTIAL,SFTINP
 106=1          COMMON /ERROR/ APBCSF,ASBCSF,BHMHSF,BHMLSF,BHMRSF,BTRNSF
 107=1          COMMON /ERROR/ PAMHSF,PAMLSF,PFMHSF,PFMLSF,SFMHS  SFMLSF
 108=1          COMMON /ERROR/ SAMHSF,SAMLSF,ENGTSF,PHOTSF,SHOTSF
 109=1          COMMON /ERROR/ MOOPCC,MLTSSF,RENTRY
 110=1          COMMON /ERROR/ NOPAN,F40PM,F80PM,F120PM,FTIME,NCLTCH
 111=1          COMMON /ERROR/ AENSF,APSSF,APMSF,ASSSF,ASMSF,APWSF,ASWSF
 112=1          COMMON /SINOUT/ IAPSEN,IASSEN,IBHMSP,IBHMSR,IPAMPR,IPFMPR
```

```
        COMMON /CINOUT/ ISFMPR,ISAMPR,IENCTM,IPHOTM,ISHOTM,IDTRNR
        COMMON /CINOUT/ TRNDIR,DIGIN,DIGOUT,SUSMSG,NSUSMG
        COMMON /CINOUT/ LAND,TRNSTN,SEA,PARK,NTRAL,REVRSE,DRIVE,HIGH,LOW
        COMMON /CINOUT/ TEST1,TEST2,TEST3,BYBPSS,HYNBPS,ENGNON,ENGMSG
        COMMON /CINOUT/ CKPUMP,CKRAMP,CKPLST,CKRLST
        COMMON /CINOUT/ CKPSLT,CKRSL1,CKRSL2
        COMMON /CINOUT/ IAENSP,IAPMSP,IASMSP,IAPSSP,IASSSP,IAPWSP,IASWSP
        COMMON /CINOUT/ DTRST,PRMOOP,ERRDLY,DTDGS,PDTDGS,PTDGS
        COMMON /CINOUT/ TDELAY,TIMER,DMOOP,PDMOOP
        COMMON /CINOUT/ SECPOF,PCPWNP,BLGPON
        COMMON /CINOUT/ SCTD,GRCLSE,GOTD,GROPEN,PCPWDP,BLGOFF,SLOWER
        COMMON /CINOUT/ SUDOWN,BLCVV,BHCVV,MNSPON,MBPON,TRKSTP,SURISE
        COMMON /CINOUT/ PATVV,PFTVV,SATVV,SFTVV
        COMMON /CINOUT/ AUX1,AUX2,AUX3,AUX4,AUX5,AUX6,AUX7,AUX8,IDES
        COMMON /CINOUT/ RSMS,RPMS,RSSS,RPSS,RES
        COMMON /CINOUT/ PTVV,STVV
        COMMON /CALC/ APBCAN,ASBCAN,DHMSP,DHMSR,DTRNR,PAMTPR
        COMMON /CALC/ FDR,WJCNST
        COMMON /CALC/ PFMTPR,SFMTPR,SAMTPR,ENCTEM,PHOTEM,SHOTEM
        COMMON /CALC/ DPMSP,DSMSP,ODHMSP,PNTIG,AUX9,AUX10
        COMMON /CALC/ DHWSP,DHWSR,DPWSP,DSWSP,INTCPT
        COMMON /CALC/ APPS,ASPS,AENSP,PMDIS,PMDFP
        COMMON /CALC/ PMTRQ,DPMP,REPPT,PTREFF,REPST
        COMMON /CALC/ APMSP,ASMSP,SMDIS,SMDFP
        COMMON /CALC/ SMTRQ,DSMP,STREFF,REPT
        COMMON /CALC/ DPPRT,PDIFP,DPPWJ,DSPRT,SDIFP,DPSWJ
        COMMON /CALC/ PPDIS,PPDFP,APSSP,ASSSP
        COMMON /CALC/ PPTRQ,DPPP,REPPP,PPMEFF,REPSP
        COMMON /CALC/ SPDIS,SPDFP,MAXMSP,TREF,DES
        COMMON /CALC/ SPTRQ,DSPP,SPMEFF,REPP,ALPMSP,ALSMSP
        COMMON /CALC/ ALPWSP,ALSWSP,MAXWSP,AUXPOW,TRNPOW,TORQUE
        COMMON /CALC/ K1,K2,K3,K4,K5,K6,K7,K8,K9,K10
        COMMON /CALC/ DHMSPG,PAMPR,SAMPR,PFMPR,SFMPR
        COMMON /MOUT/ M1,M2,M3,M4,M5,M6,M7,M8,M9,M10
        COMMON /MOUT/ M11,M12,M13,M14,M15,M16,M17,M18,M19,M20
        COMMON /MOUT/ M21,M22,M23,M24,M25,M26,M27,M28,M29,M30
        COMMON /MOUT/ M31,M32,M33,M34,M35,M36,M37,M38,M39,M40
        COMMON /MOUT/ M41,M42,M43,M44,M45,M46,M47,M48,M49,M50
C
C       TEST TO SEE IF THE SHIFT IS ALLOWABLE
C
        IF(.NOT.FTIME) THEN
             FFDR=4.43
             DPMSP=APMSP
             DSMSP=ASMSP
             CALL MOTREP
             DPMSP=APSSP*FFDR
             DSMSP=ASMSP*FFDR
C
C       MOTREP GAVE US THE PRESENT MOTOR POWER, NOW CALC REQUIRED TORQUE
C
             FPMD=APPS*12.89/APMSP
             FSMD=ASPS*12.89/ASMSP
             IF(FPMD.GT.11.36) FPMD=11.36
             IF(FSMD.GT.11.36) FSMD=11.36
             PMDFP=PMDIS*PMDFP*2.34/FPMD
             SMDFP=SMDIS*SMDFP*2.34/FSMD
             IF((PMDFP.GT.5000).OR.(SMDFP.GT.5000)) THEN
```

```
                  SFTINP=.FALSE.
                  RETURN
               END IF
      C
      C    DISENGAGE CLUTCHES
      C
               FTIME=.TRUE.
               BLCVV=0
               BHCVV=0
           ENDIF
      C
      C    CHECK TO SEE IF THE CLUTCHES ARE SYNCHED
      C
           FBR=4.48
           ALPMSP=APSSP*FBR
           ALSMSP=ASSSP*FBR
           IF((ABS(ALPMSP-APMSP).LT.400).AND.(ABS(ALSMSP-ASMSP).LT.400)) THEN
               BHCVV=65535
               BLCVV=0
               SFTINP=.FALSE.
               PTBSS=PLTBSS
               NCLTCH=.FALSE.
               FTIME=.FALSE.
           ENDIF
      C
           RETURN
           END
```

```
FORTRAN-86 COMPILER        ERROR MESSAGES AND SUMMARY
:F2:UPSHFT.FOR

STORAGE REQUIREMENTS FOR MODULE UPSHFT:

     CODE AREA SIZE        002CFH        719D
     CONSTANT AREA SIZE    00022H         34D
     VARIABLE AREA SIZE    00010H         16D
     MAXIMUM STACK SIZE    00012H         18D
     /ERROR/               0012CH        300D
     /MOUT/                000C6H        198D
     /CINOUT/              0019CH        412D
     /BITFNC/              00268H        616D
     /CALC/                0020CH        524D

0 ERRORS DETECTED.
0 WARNINGS ISSUED.
ENTRY POINT IS 24H
FLOATING-POINT OPERATIONS WERE GENERATED.
COMPILATION OF UPSHFT COMPLETE.


0 TOTAL ERRORS DETECTED.
0 TOTAL WARNINGS ISSUED.
END OF FORTRAN-86 COMPILATION.
```

# Appendix B

## Individual Circuit Diagrams

Input/Output Pin Sheets for SC-1 Microcomputer for ATR Vehicle

| | |
|---|---|
| J7 | Pulse Width Modulated (PWM) Output Signal |
| P1 | Desired Low Clutch Valve Voltage (DLCVV) (Page B-9) |
| P2 | Desired High Clutch Valve Voltage (DHCVV) (Page B-10) |
| P3 | |
| P4 | |
| P5 | Ground (Page B-11) |
| J8 | Pulse Width Modulated (PWM) Output Signal |
| P1 | Port Forward Transmission Valve Voltage (PFTVV) (Page B-12) |
| P2 | Starboard Forward Transmission Valve Voltage (SFTVV) (Page B-13) |
| P3 | Port Aft Transmission Valve Voltage (PATVV) (Page B-14) |
| P4 | Starboard Aft Transmission Valve Voltage (SATVV) (Page B-15) |
| P5 | Ground (Page B-16) |
| J9 | Digital to Analog Converter |
| P1 | |
| P4 | |
| P5 | |
| P9 | |
| P10 | Read Port Sprocket Speed (RPSS) (Page B-17) |
| P11 | |
| J11 | Frequency to Digital (F/D) Converter |
| P11 | Ground (Page B-18) |
| P16 | Input Actual Port Motor Speed (IAPMSP) (Page B-19) |
| P17 | Input Actual Starboard Motor Speed (IASMSP) (Page B-20) |
| P18 | Input Actual Port Sprocket Speed (IAPSSP) (Page B-21) |
| P19 | Input Actual Starboard Sprocket Speed (IASSSP) (Page B-22) |
| J12 | Frequency to Digital (F/D) Converter |
| P11 | Ground (Page B-23) |
| P16 | Input Actual Port Waterjet Speed (IAPWSP) (Page B-24) |

| | |
|---|---|
| P17 | Input Actual Starboard Waterjet Speed (IASWSP) (Page B-25) |
| P18 | Input Actual Engine Speed (IAENSP) (Page B-26) |
| P19 | |
| J13 | Analog to Digital (A/D Converter |
| P1 | Input Desired High Motor Speed (IDHMSP) (Page B-27) |
| P2 | Input Desired High Motor Speed Ratio (IDHMSR) (Page B-28) |
| P3 | Input Desired Turn Ratio (IDTRNR) (Page B-29) |
| P4 | Input Port Bucket Angle (IAPBCN) (Page B-30) |
| P5 | Input Starboard Bucket Angle (IASBCN) (Page B-31) |
| P6 | |
| P7 | |
| P8 | |
| P9 | Ground (Page B-32) |
| P10 | Ground (Page B-33) |
| P11 | Ground (Page B-34) |
| P12 | Ground (Page B-35) |
| P13 | |
| P14 | |
| P15 | |
| P16 | Ground (Page B-36) |
| P17 | Computer Ground (Page B-37) |
| P18 | Computer Ground (Page B-37) |
| J14 | Analog to Digital (A/D) Converter |
| P1 | Input Port Forward Motor Pressure (IPFMPR) (Page B-38) |
| P2 | Input Starboard Forward Motor Pressure (ISFMPR) (Page B-39) |
| P3 | Input Port Aft Motor Pressure (IPAMPR) (Page B-40) |
| P4 | Input Starboard Aft Motor Pressure (ISAMPR) (Page B-41) |
| P5 | Input Port Hydraulic Oil Temperature (IPHOTM) (Page B-42) |
| P6 | Input Starboard Hydraulic Oil Temperature (ISHOTM) (Page B-43) |
| P7 | Input Engine Coolant Temperature (IENCTM) (Page B-44) |
| P8 | |
| P9 | Ground (Page B-45) |
| P10 | Ground (Page B-46) |

| | |
|---|---|
| P11 | Ground (Page B-47) |
| P12 | Ground (Page B-48) |
| P13 | Ground (Page B-49) |
| P14 | Ground (Page B-50) |
| P15 | Ground (Page B-51) |
| P16 | |
| P17 | Computer Ground (Page B-52) |
| P18 | Computer Ground (Page B-52) |

| | |
|---|---|
| J16 | Parallel I/O (Switch Closures to Ground) |
| P1 | Clamp (+24) (Page B-53) |
| P18 | Desired Suspension Up Valve Status (Page B-60) |
| P17 | Desired Suspension Down Valve Status (Page B-59) |
| P16 | Desired Secondary Cooling Fan 4 gpm Valve Status (Page B-58) |
| P15 | Desired Secondary Cooling Fan 8 gpm Valve Status (Page B-57) |
| P14 | Desired Electric Bilge Pump Activation (Page B-56a and B-56b) |
| P13 | |
| P12 | Desired Starboard Bucket Counterclockwise Valve Status (Page B-55) |
| P11 | Desired Starboard Bucket Clockwise Valve Status (Page B-54) |
| P10 | Clamp (+24) (Page B-53) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |
| P4 | |
| P3 | |
| P2 | |
| P20 | Clamp (+24) (Page B-53) |
| P28 | Low Brake and Clutch Lube Pressure Switch Status (Page B-69) |
| P27 | Low Brake Release Pressure Switch Status (Page B-68) |
| P26 | Low Brake Supply Pressure Switch Status (Page B-67) |
| P25 | Low Port Charge Pump Pressure Switch Status (Page B-66) |
| P24 | Low Starboard Charge Pump Pressure Switch Status (Page B-65) |

| | |
|---|---|
| P23 | Hydraulic Filter By-Pass Switch Status (Page B-64) |
| P22 | Port 1 Suspension Component Switch Status (Page B-63) |
| P21 | Port 2 Suspension Component Switch Status (Page B-62) |
| P19 | Port 3 Suspension Component Switch Status (Page B-61) |
| P29 | |
| P36 | Port 4 Suspension Component Switch Status (Page B-72) |
| P35 | Port 5 Suspension Component Switch Status (Page B-71) |
| P34 | Low Suspension System Supply Pressure (Page B-70) |
| P33 | |
| P32 | |
| P31 | |
| P30 | |
| P37 | Signal Return for Computer (Page B-73) |
| | |
| J17 | Parallel I/O (Switch Closure to Ground) |
| P1 | Clamp (+24) (Page B-74) |
| P18 | |
| P17 | Desired Primary Cooling Fan on Valve Status (Page B-80) |
| P16 | Desired Primary Cooling Fan Stop Valve Status (Page B-79) |
| P15 | Desired Secondary Grill Open Valve Status (Page B-78) |
| P14 | |
| P13 | Desired Seawater Pump Activation (Page B-77) |
| P12 | Desired Waterjet By-Pass Valve Status (Page B-76) |
| P11 | Desired Waterjet No By-Pas Valve Status (Page B-75) |
| P10 | Clamp (+24) (Page B-74) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |
| P4 | |
| P3 | |
| P2 | |
| P20 | (Clamp (+24) (Page B-74) |
| P28 | Starboard 1 Suspension Component Switch Status (Page B-88) |

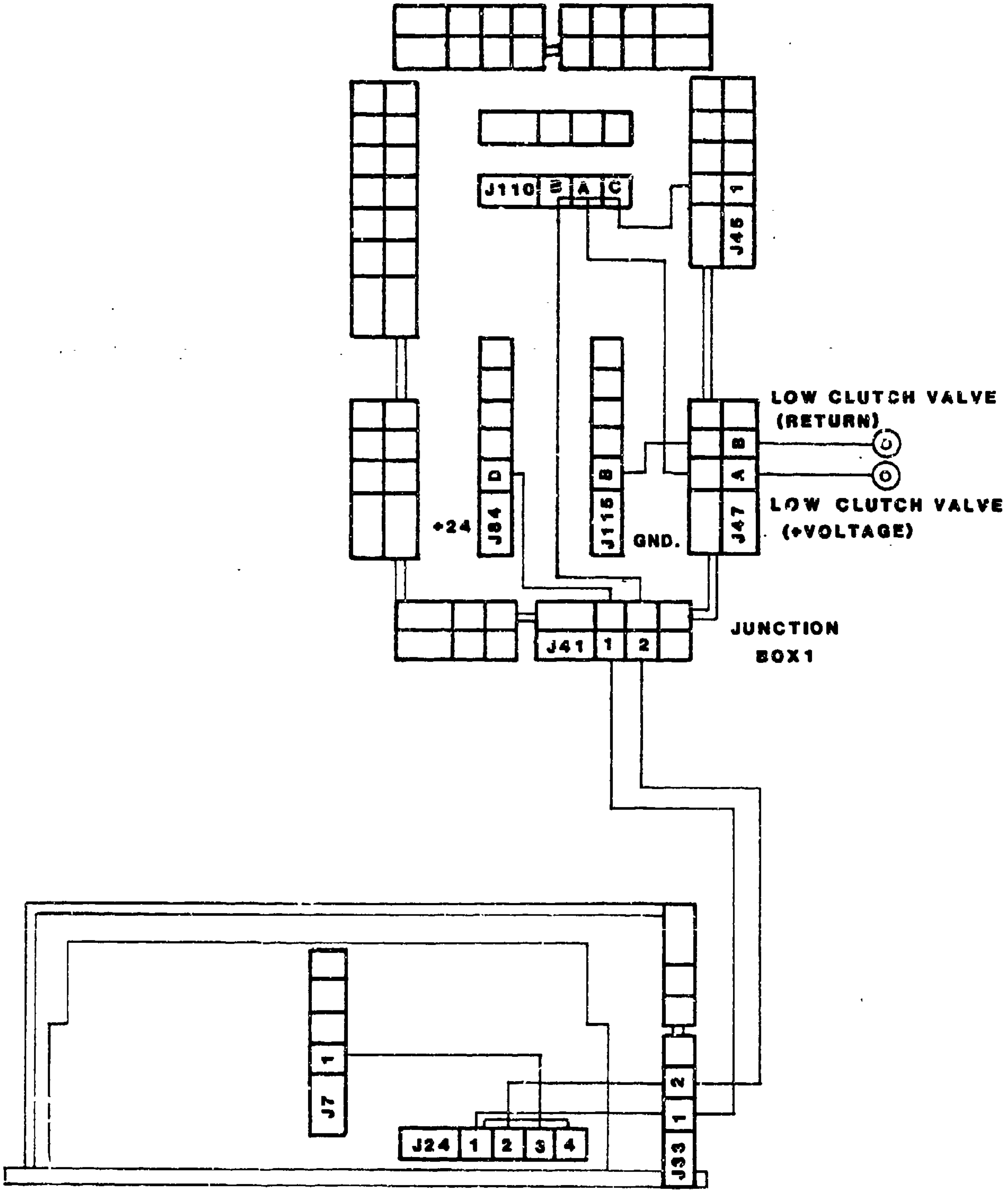
| | |
|---|---|
| P27 | Starboard 2 Suspension Component Switch Status (Page B-87) |
| P26 | Starboard 3 Suspension Component Switch Status (Page B-86) |
| P25 | Starboard 4 Suspension Component Switch Status (Page B-85) |
| P24 | Starboard 5 Suspension Component Switch Status (Page B-84) |
| P23 | Low Engine Oil Pressure Switch Status (Page B-83) |
| P22 | Low Port Hydraulic Oil Level Switch Status (Page B-82) |
| P21 | Low Starboard Hydraulic Oil Level Switch Status (Page B-81) |
| P19 | |
| P29 | Cathode Signal Return (Page B-89) |
| P36 | Low Vehicle System Voltage Switch Status (Page B-95) |
| P35 | Low Control System Voltage Switch Status (Page B-94) |
| P34 | High Vehicle System Voltage Switch Status (Page B-93) |
| P33 | High Control System Voltage Switch Status (Page B-92) |
| P32 | Primary Grill Closed Switch Status (Page B-91) |
| P31 | Ramp Closed and Latched Switch Status (Page B-90) |
| P30 | |
| P37 | Signal Return for Computer (Page B-96) |
| J18 | Parallel I/O (Positive Voltage to Card) |
| P1 | Clamp (+24) (Page B-97) |
| P18 | Desired Motor No By-Pass Valve Status (Page B-105) |
| P17 | Desired Port Bucket Counterclockwise Valve Status (Page B-104) |
| P16 | Desired Port Bucket Clockwise Valve Status (Page B-103) |
| P15 | Desired Primary Grill Closure Valve Status (Page B-102) |
| P14 | Desired Primary Grill Open Valve Status (Page B-101) |
| P13 | Desired Hydraulic Bilge Pump Valve Activation (Page B-100) |
| P12 | Desired Hydraulic By-Pass Valve Status (Page B-99) |
| P11 | Desired Motor By-Pass Val e Status (Page B-98) |
| P10 | Clamp (+24) (Page B-97) |
| P9 | |
| P8 | |
| P7 | |
| P6 | |
| P5 | |
| P4 | |

| | |
|---|---|
| P3 | |
| P2 | |
| P20 | Clamp (+24) (Page B-97) |
| P28 | Aft Electric Bilge Pump Activation Status (Page B-114) |
| P27 | Seawater Pump Activation Status (Page B-113) |
| P26 | Hydraulic Bilge Pump Valve Activation Status (Page B-112) |
| P25 | Forward Electric Bilge Pump Activation Status (Page B-111) |
| P24 | Desired High Gear Selector Switch Status (Page B-110) |
| P23 | Desired Low Gear Selector Switch Status (Page B-109) |
| P22 | Desired Seaborne Mode Selector Switch Status (Page B-108) |
| P21 | Desired Transition Mode Selector Switch Status (Page B-107) |
| P19 | Desired Landborne Mode Selector Switch Status (Page B-106) |
| P29 | Cathode Signal Return (Page B-115) |
| P36 | Desired Drive Selector Switch Status (Page B-121) |
| P35 | |
| P34 | Desired Reverse Selector Switch Status (Page B-119) |
| P33 | Desired Park Selector Switch Status (Page B-118) |
| P32 | Fire Sensed Switch Status (Page B-117) |
| P31 | Low Scavenge Pump Pressure Switch Status (Page B-116) |
| P30 | |
| P37 | Signal Return for Computer (Page B-122) |
| J28 | RS 232 Transmit (from SC-1) |
| P2 | RS 232 Output (from SC-1) Page B-123) |
| P3 | RS 232 Receive (by SC-1) (Page B-124) |
| P7 | RS232 Signal Return (Page B-125) |
| J46 | Terra Computer Power |
| E | Ground (Page B-126) |
| F | +12 Volts (Page B-126) |
| J32 | Miltope Recorder Power |
| 82 | +24 Volts (Page B-127) |
| 83 | Return (page B-127) |
| 84 | Chassis Ground (Page B-127) |

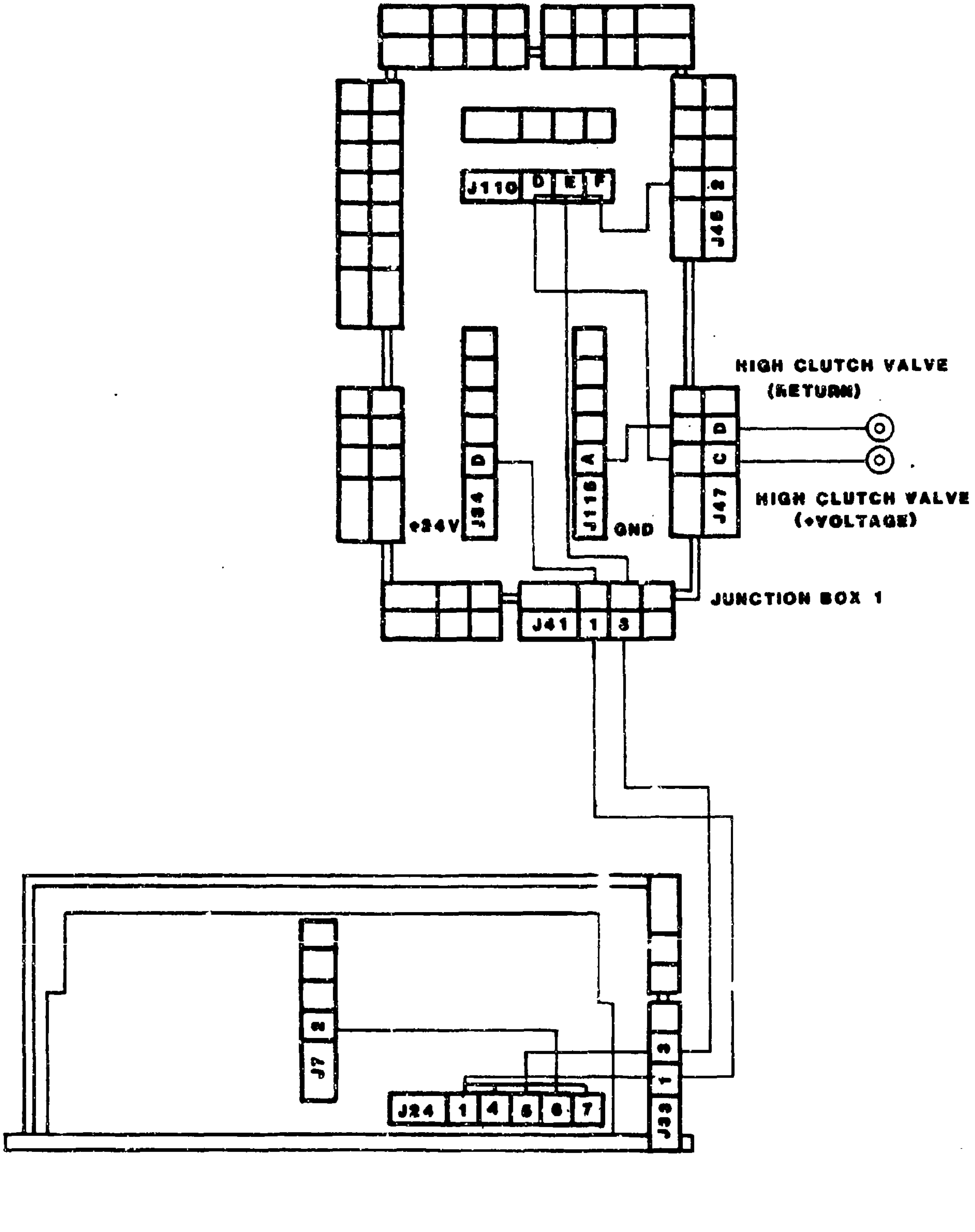
| J62 | Terra Computer Warning Signal Output |
|---|---|
| A | Light Enable (Page B-128) |
| B | Return (Page B-128) |
| C | Time Enable (Page B-128) |
| D | Return (Page B-128) |

| J40S | Terra Computer I/O to MODEM |
|---|---|
| A | Signal Return (Page B-129) |
| B | CTS from MODEM (Page B-129) |
| C | LTS to MODEM (Page B-129) |
| D | RX from MODEM (Page B-129) |
| E | TX to MODEM (Page B-129) |
| F | Ground (Page B-129) |

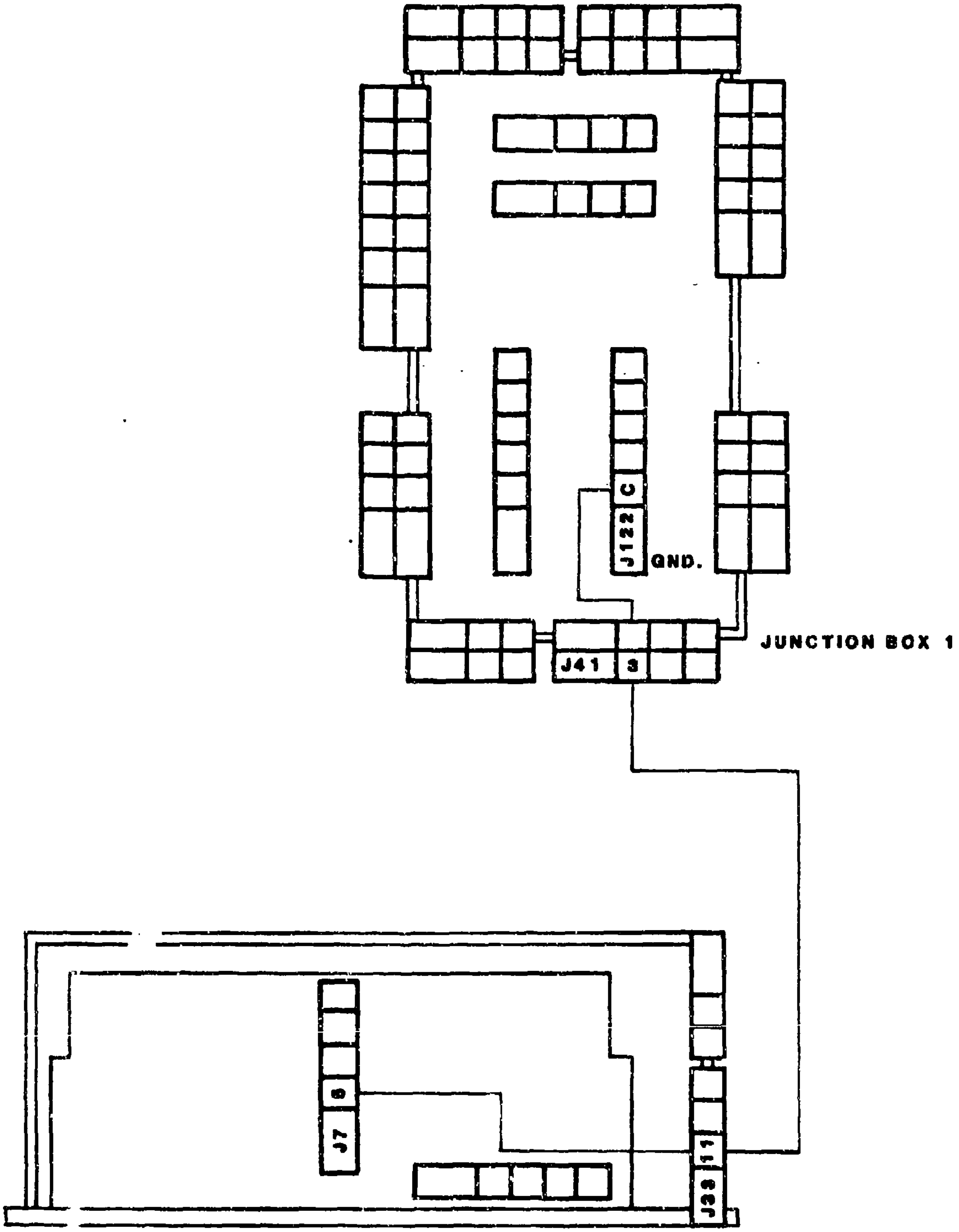
# J7-P1 LOW CLUTCH VALVE VOLTAGE



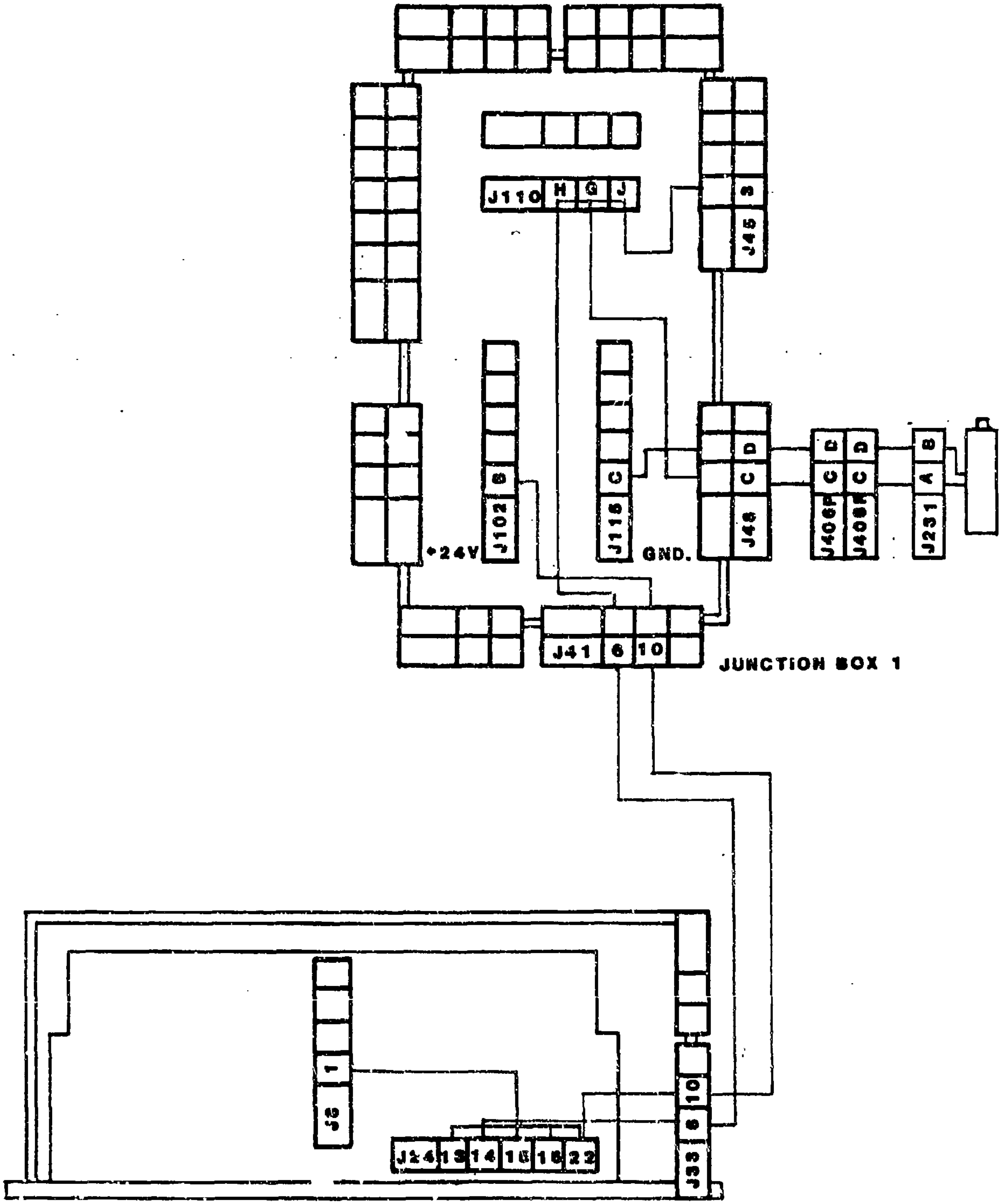
# J7-P2 HIGH CLUTCH VALVE VOLTAGE



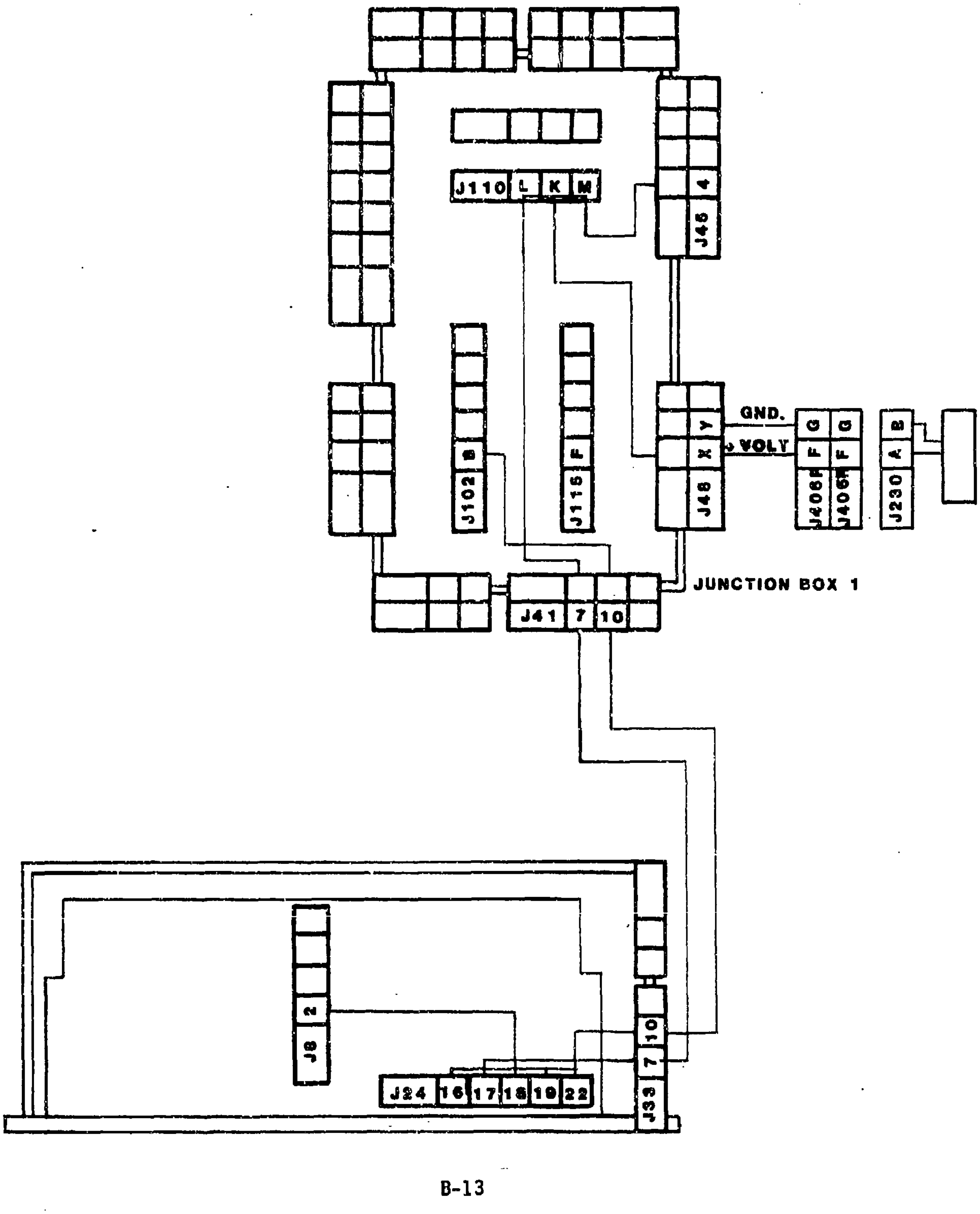
# J7-P5 GROUND



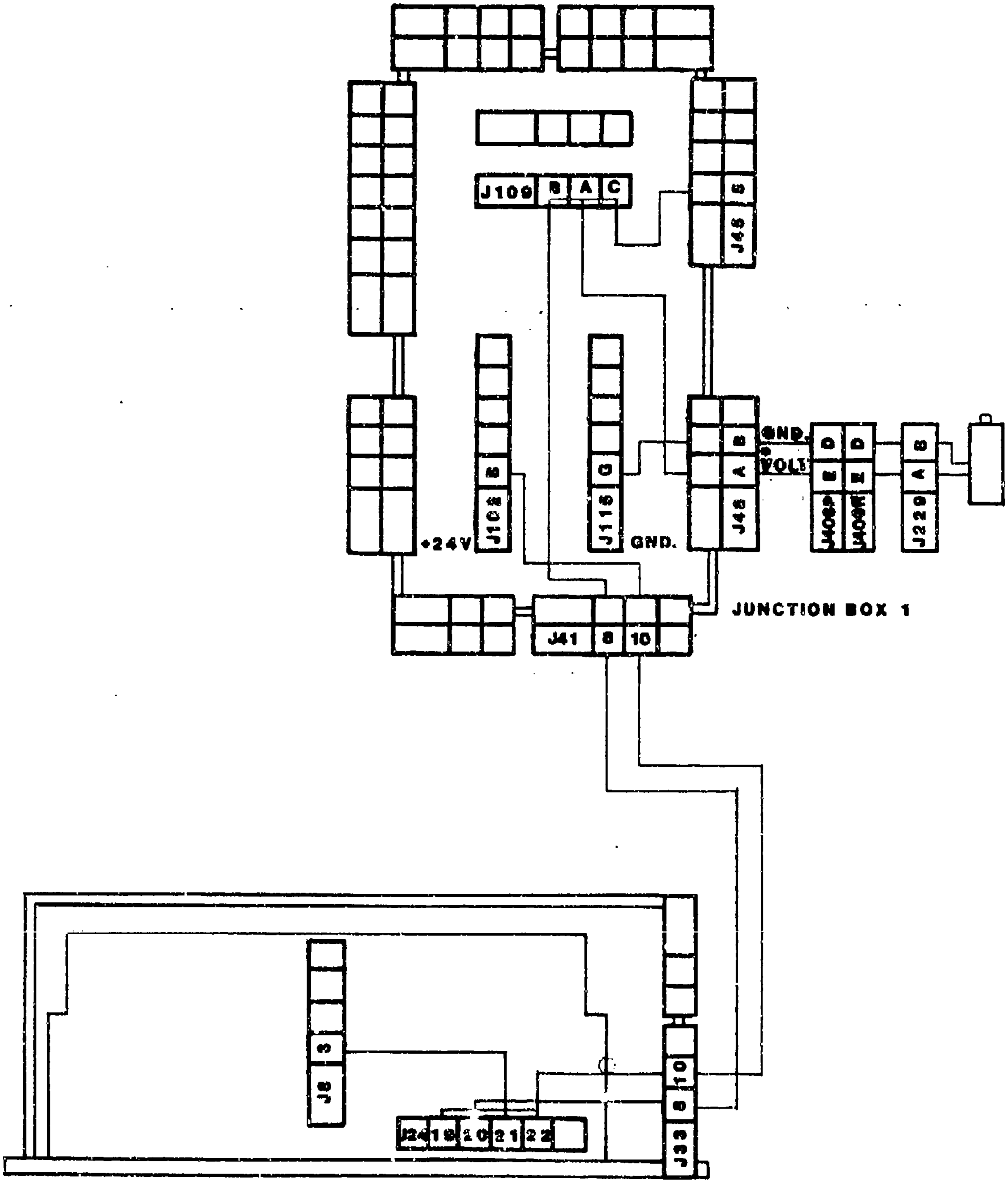
# J8-P1 PORT FORWARD TRANSMISSION VALVE VOLTAGE



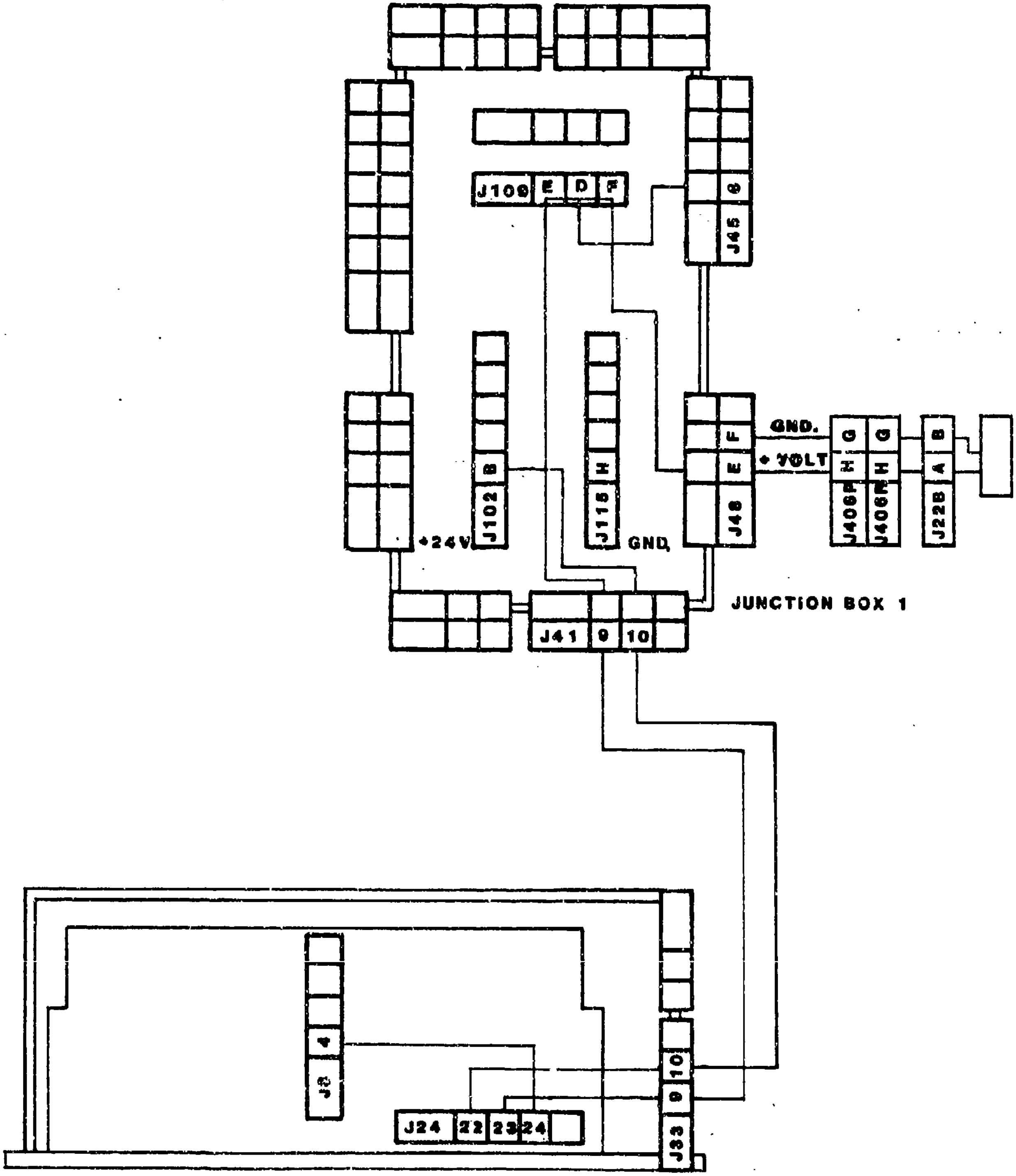
# J8-P2 STARBOARD FORWARD TRANSMISSION VALVE VOLTAGE



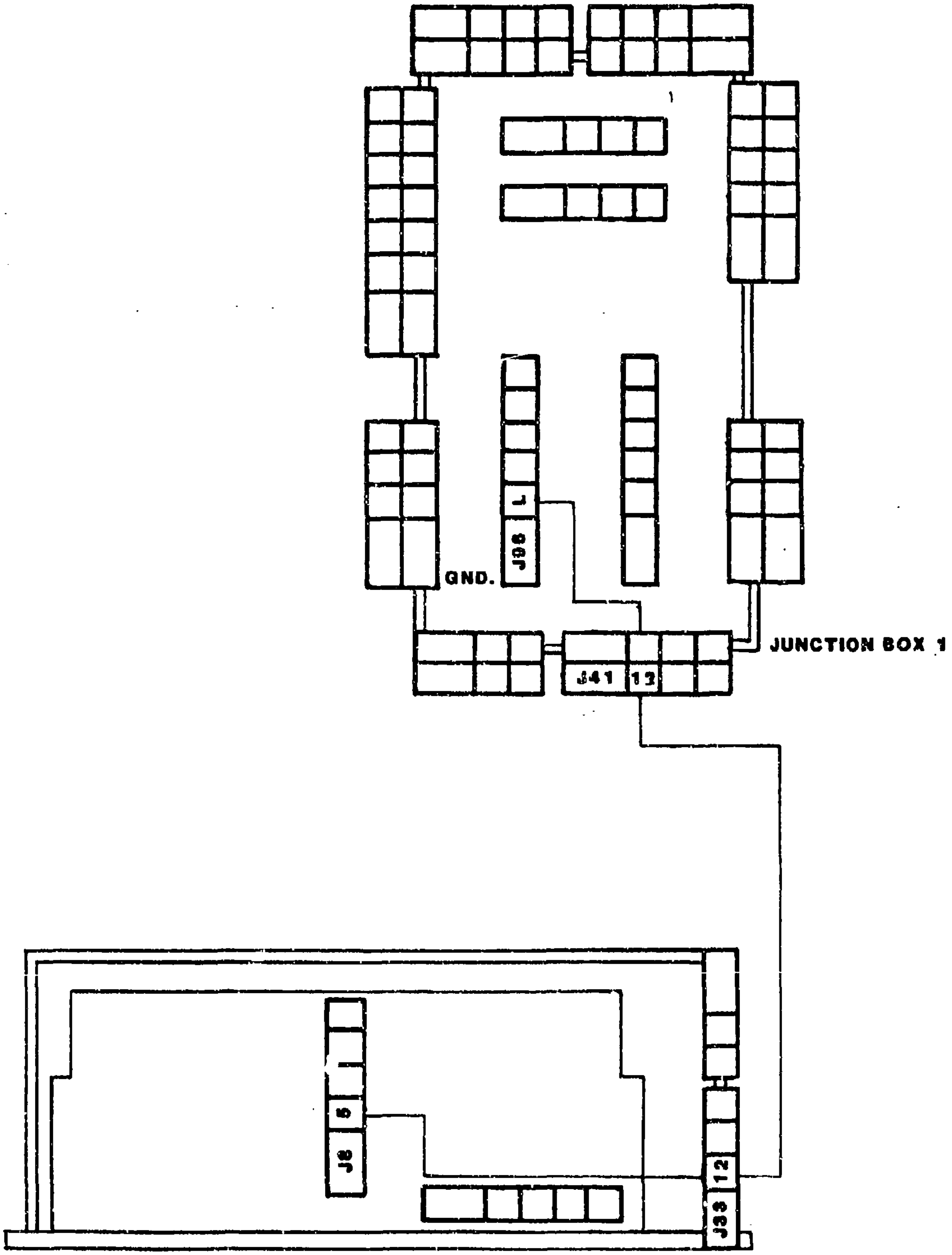
# J8-P3 PORT AFT TRANSMISSION VALVE VOLTAGE



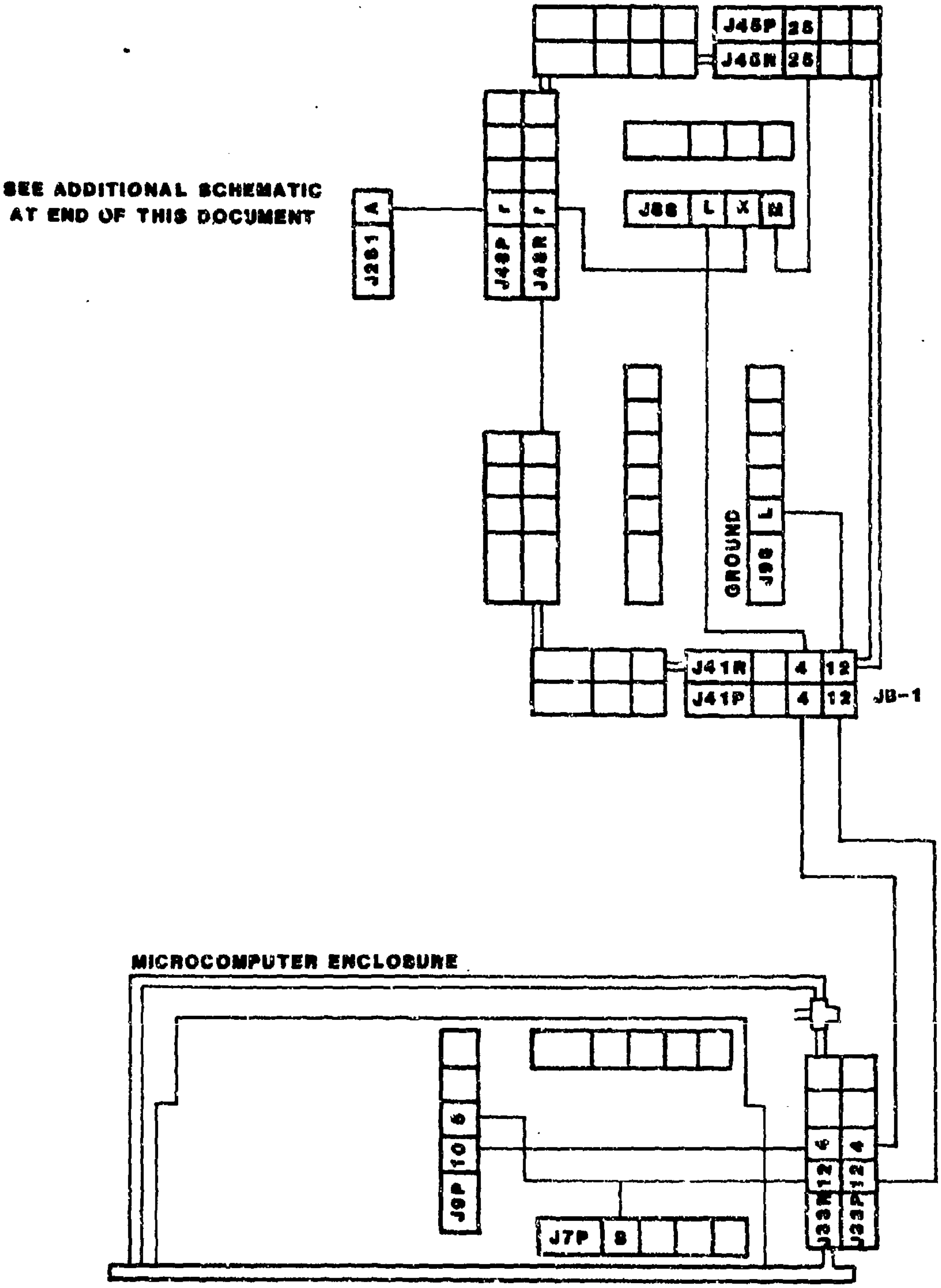
# J8-P4 STARBOARD AFT TRANSMISSION VALVE VOLTAGE



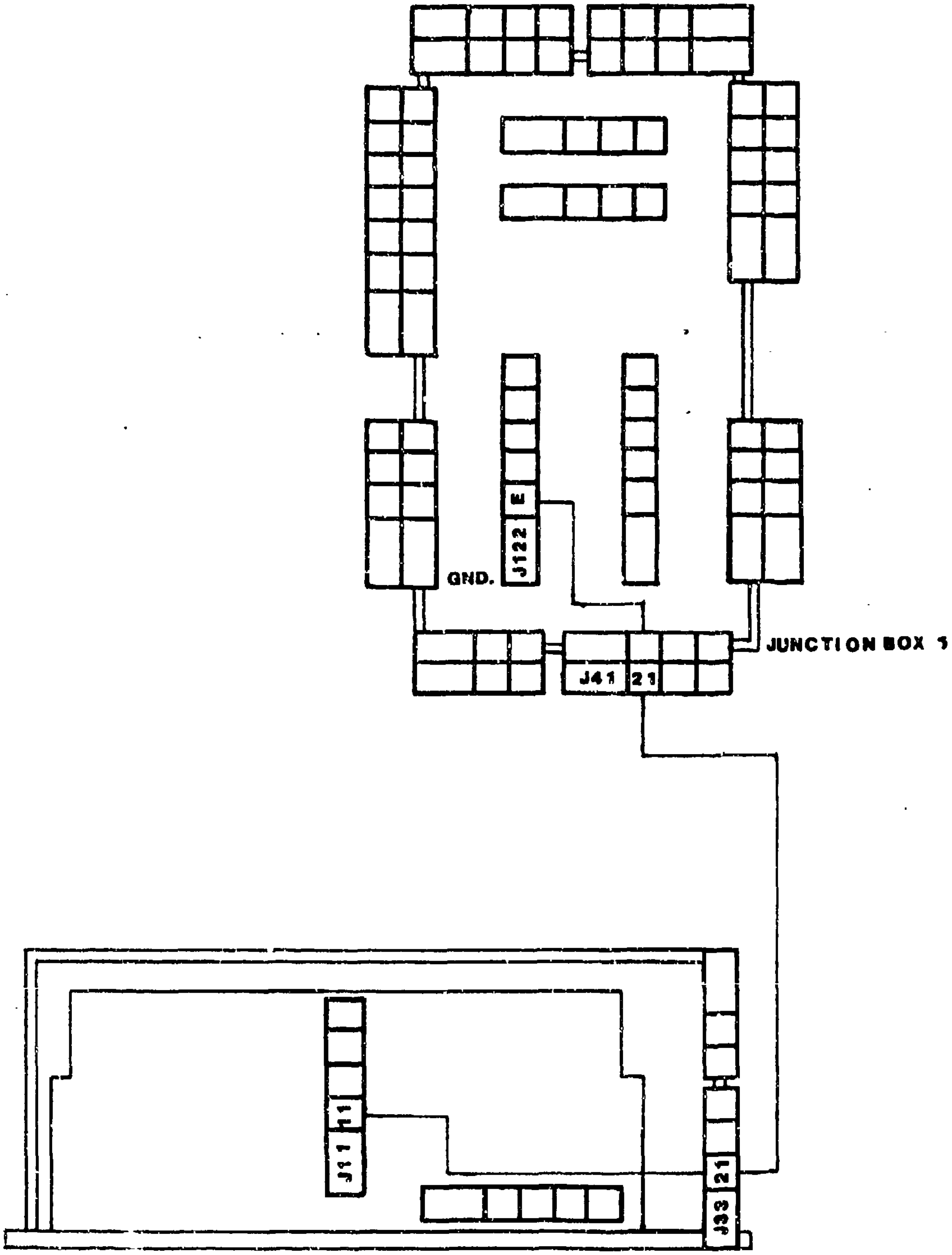
# J8-P5 GROUND



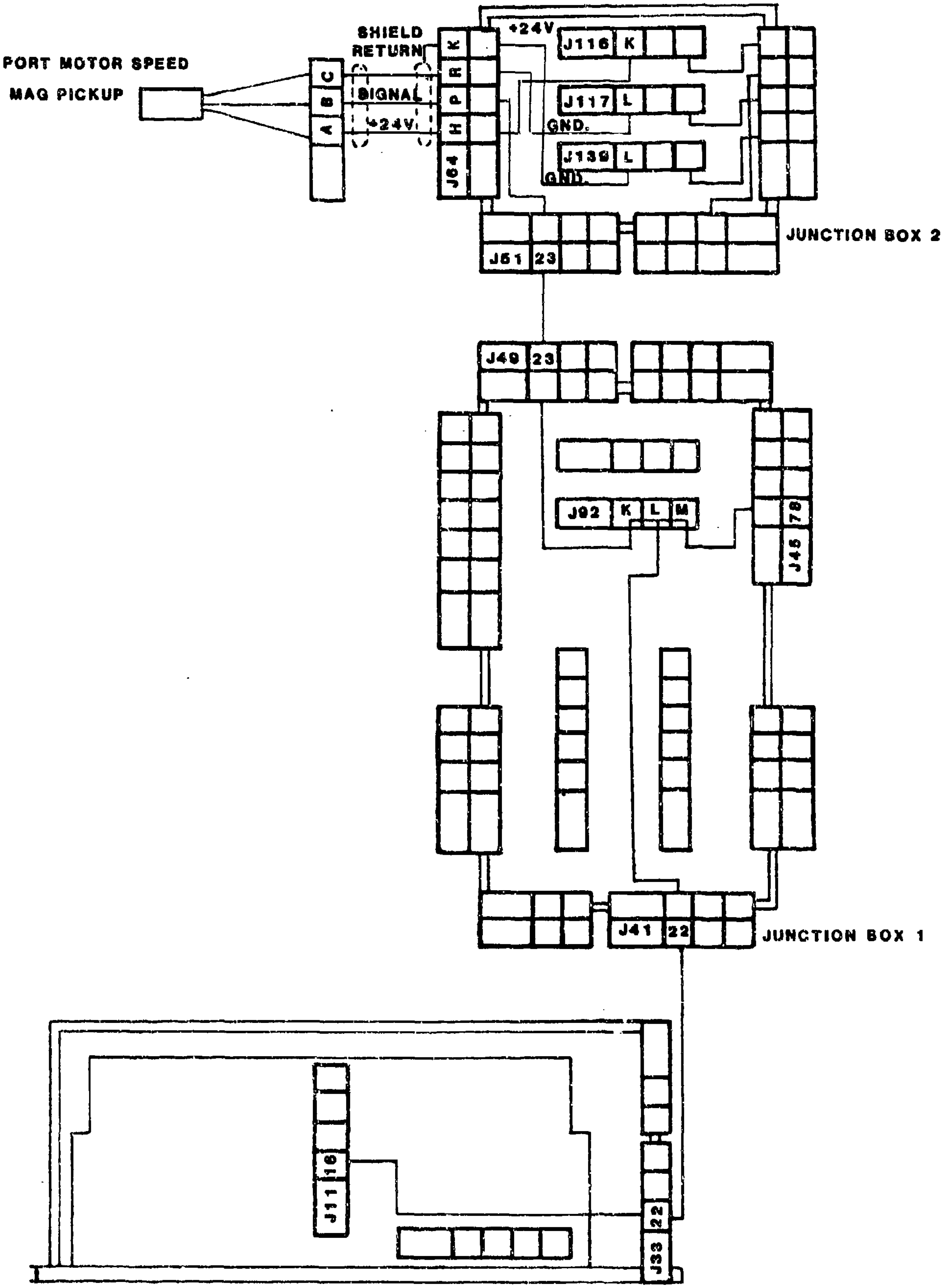
# J9P-P10 DES DESIRED ENGINE SPEED



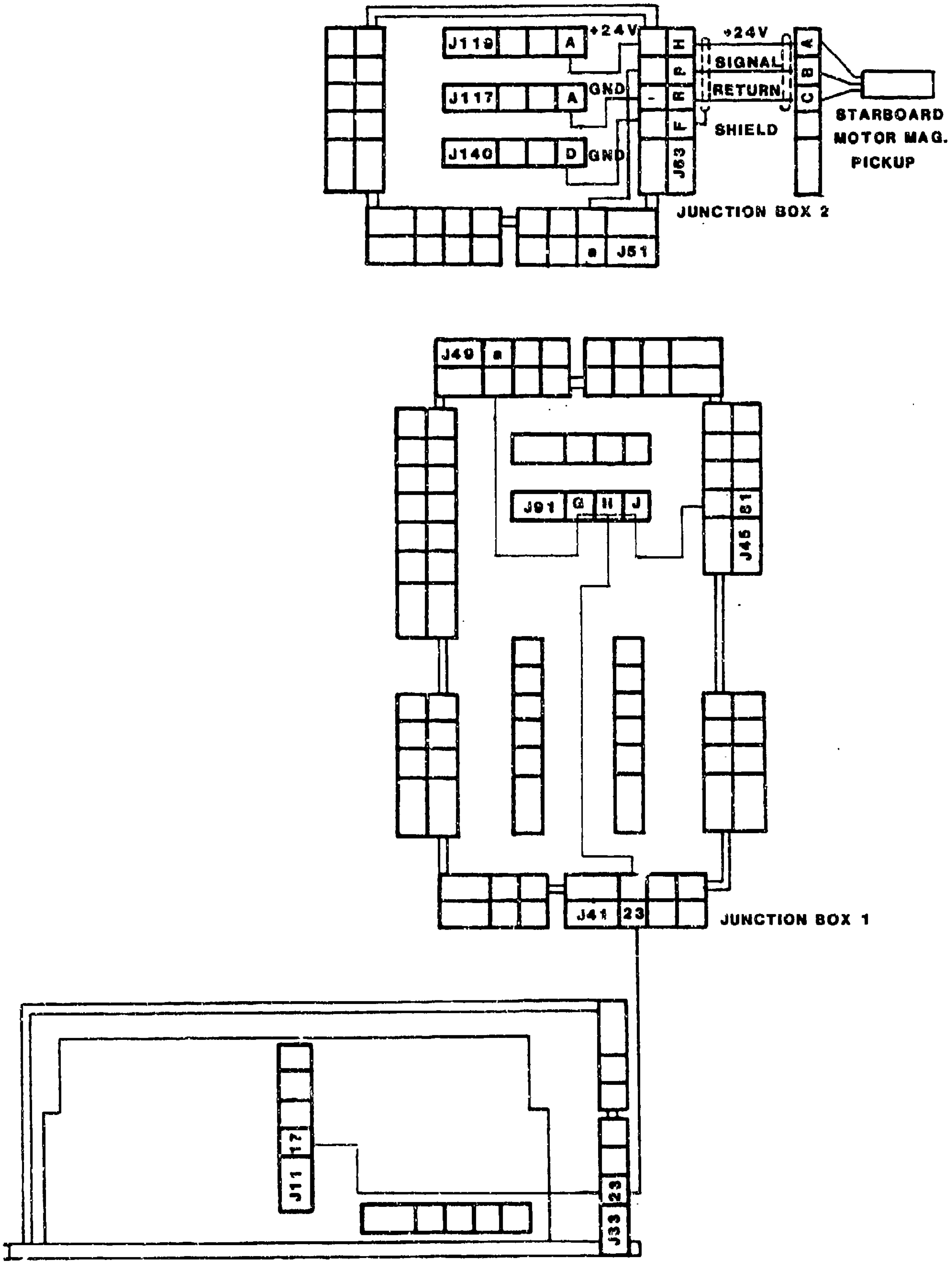
## J11-P11 GROUND



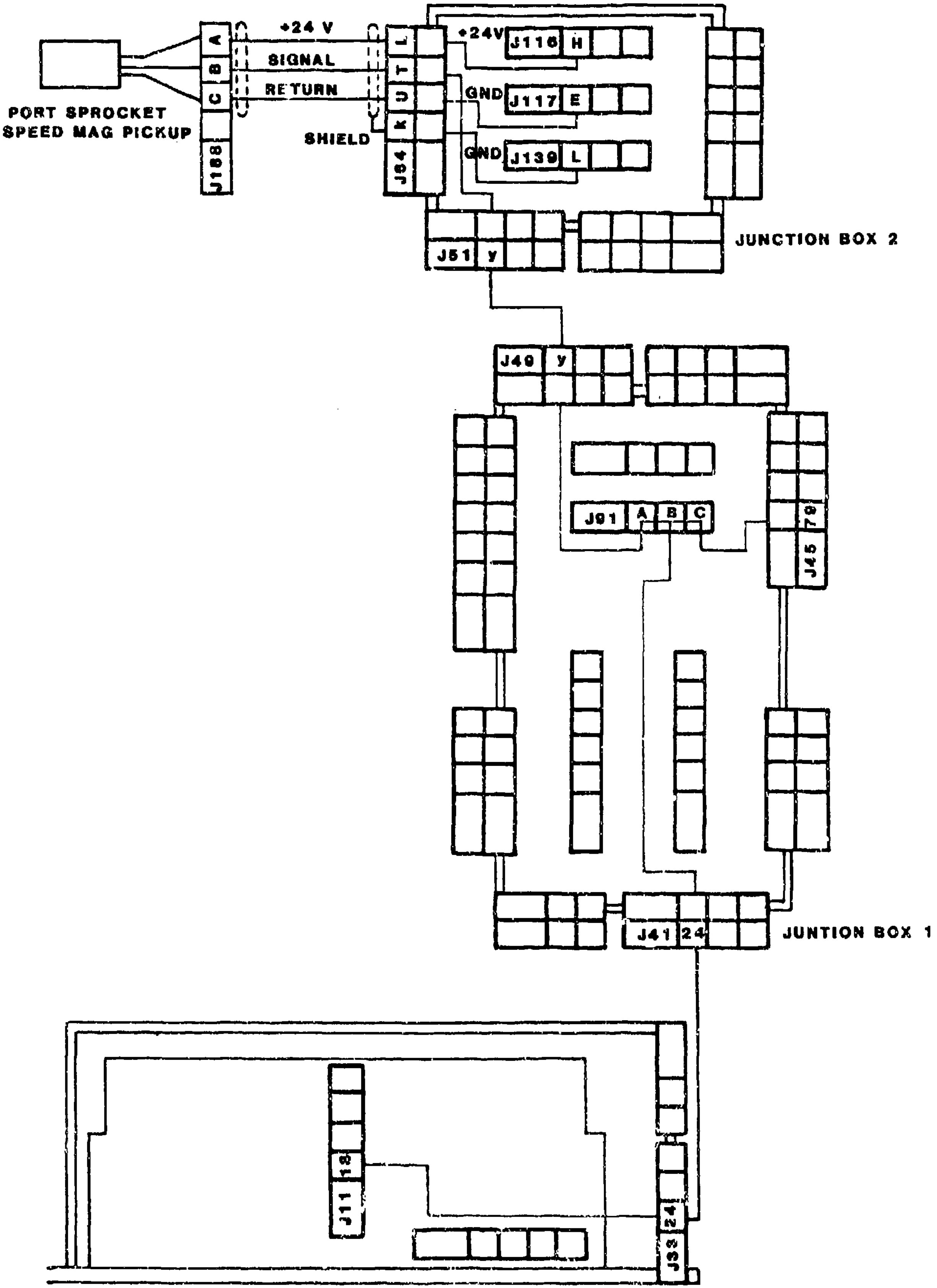
# J11-P16 INPUT PORT MOTOR SPEED



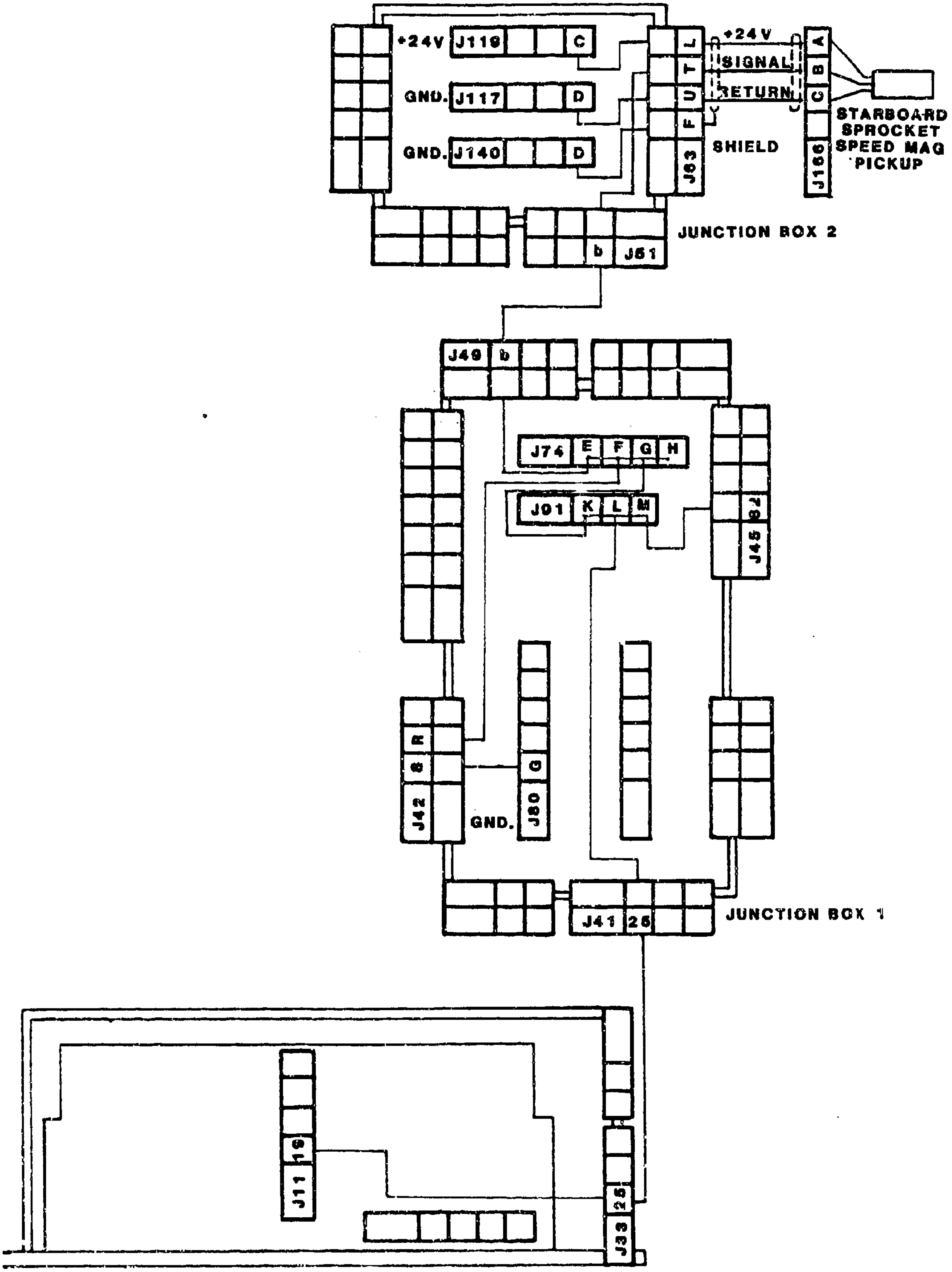
# J11-P17 INPUT STARBOARD MOTOR SPEED



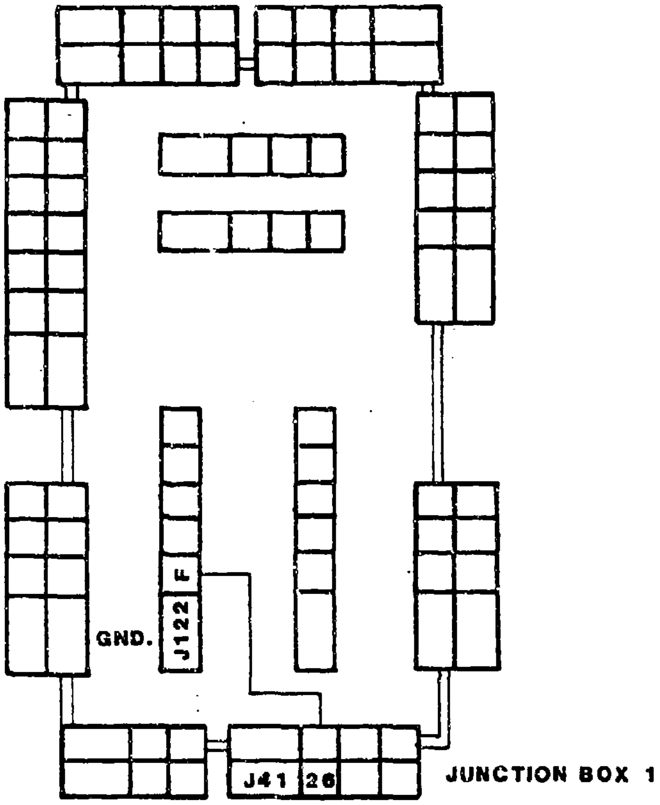
# J11-P18 INPUT PORT SPROCKET SPEED

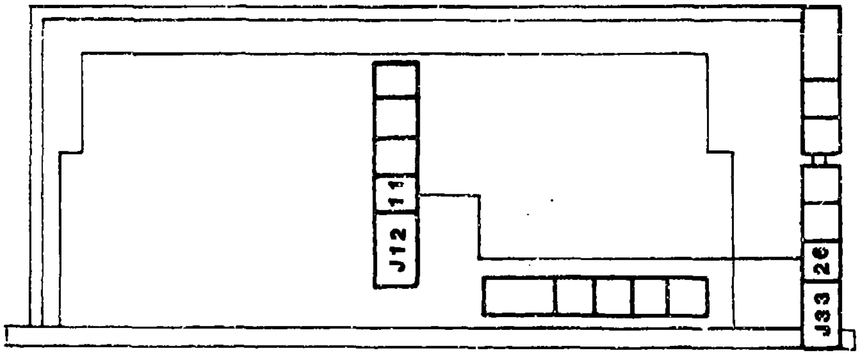


# J11-P19 INPUT STARBOARD SPROCKET SPEED

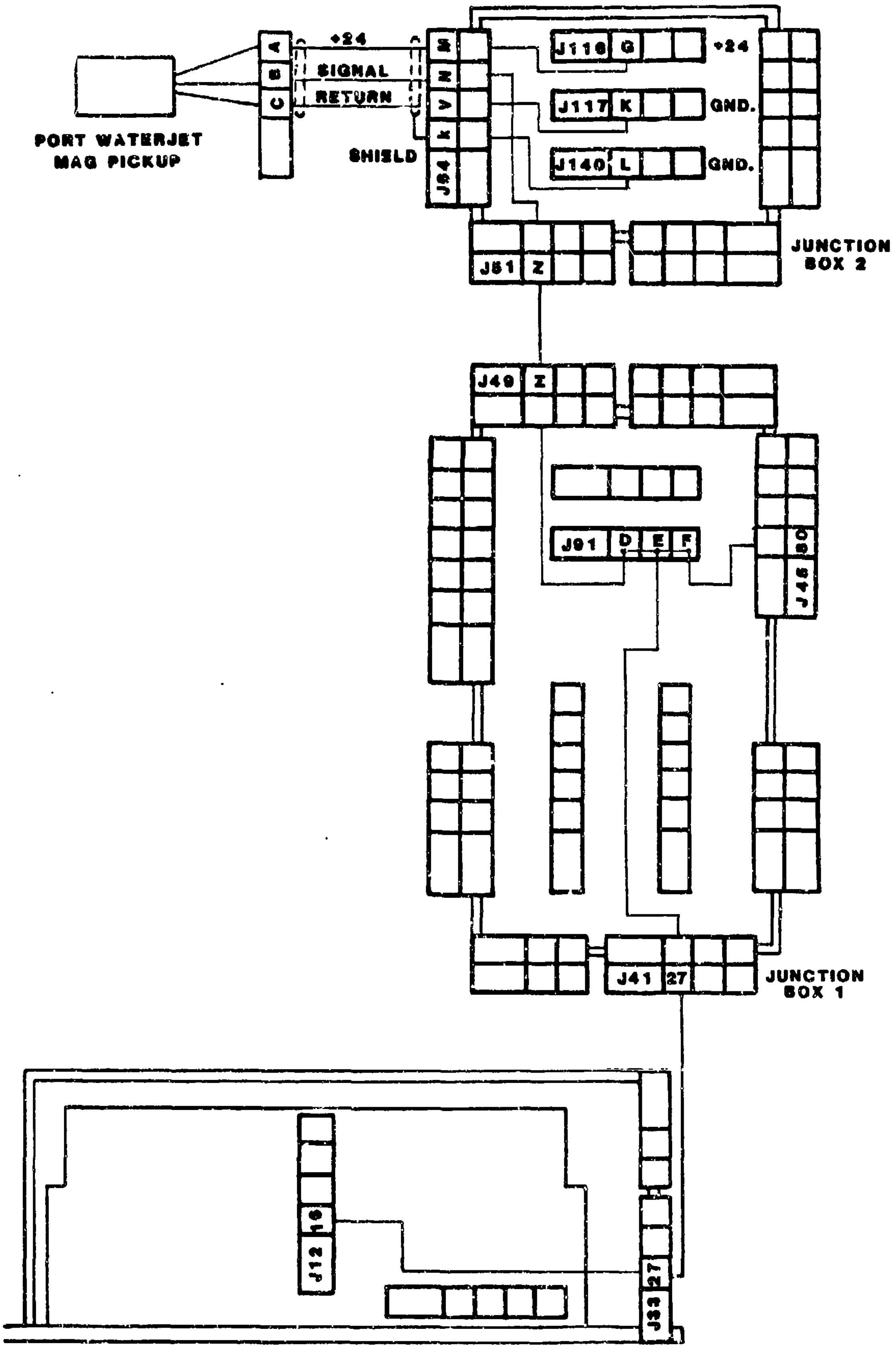


# J12-P11 GROUND

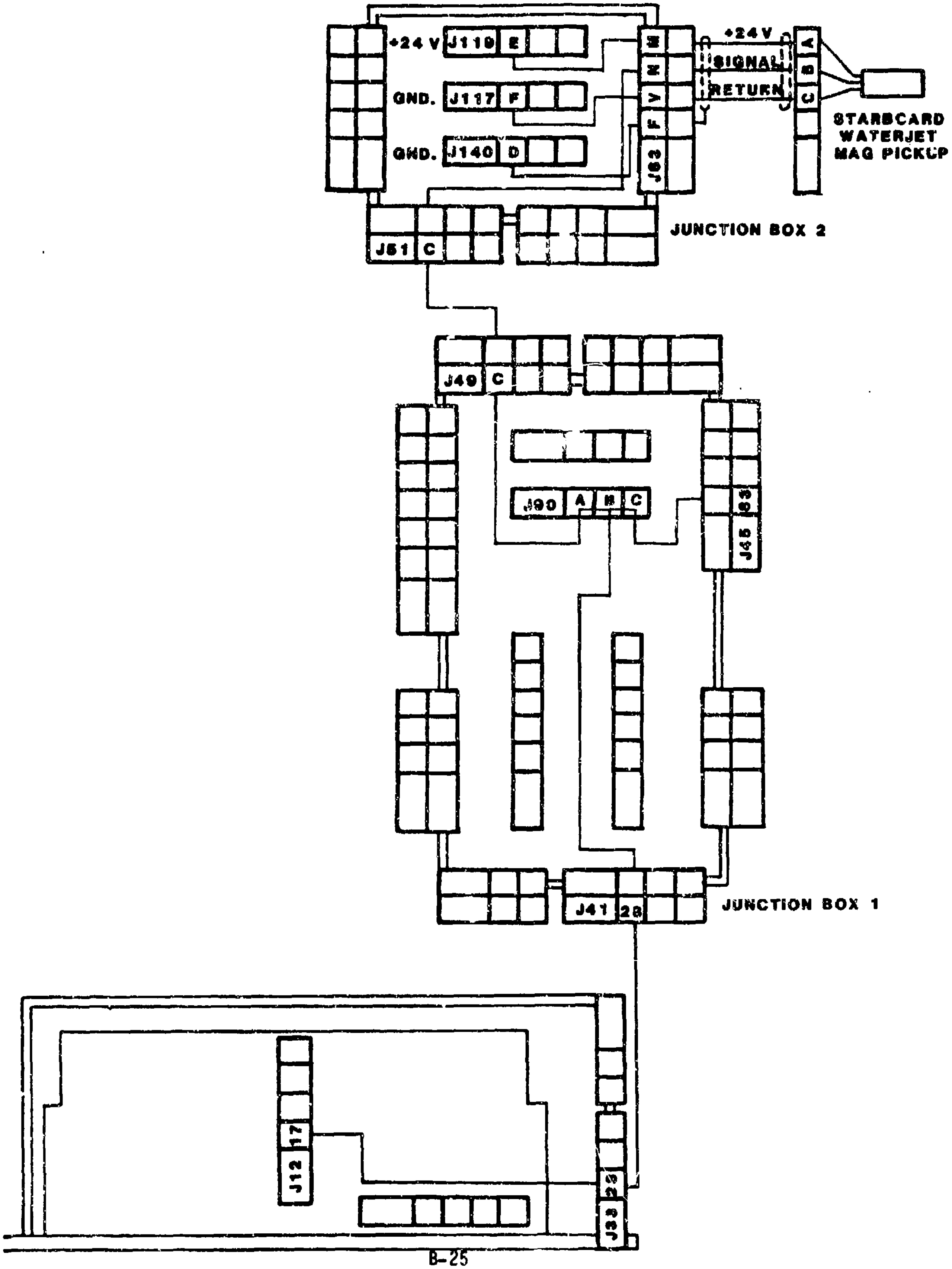




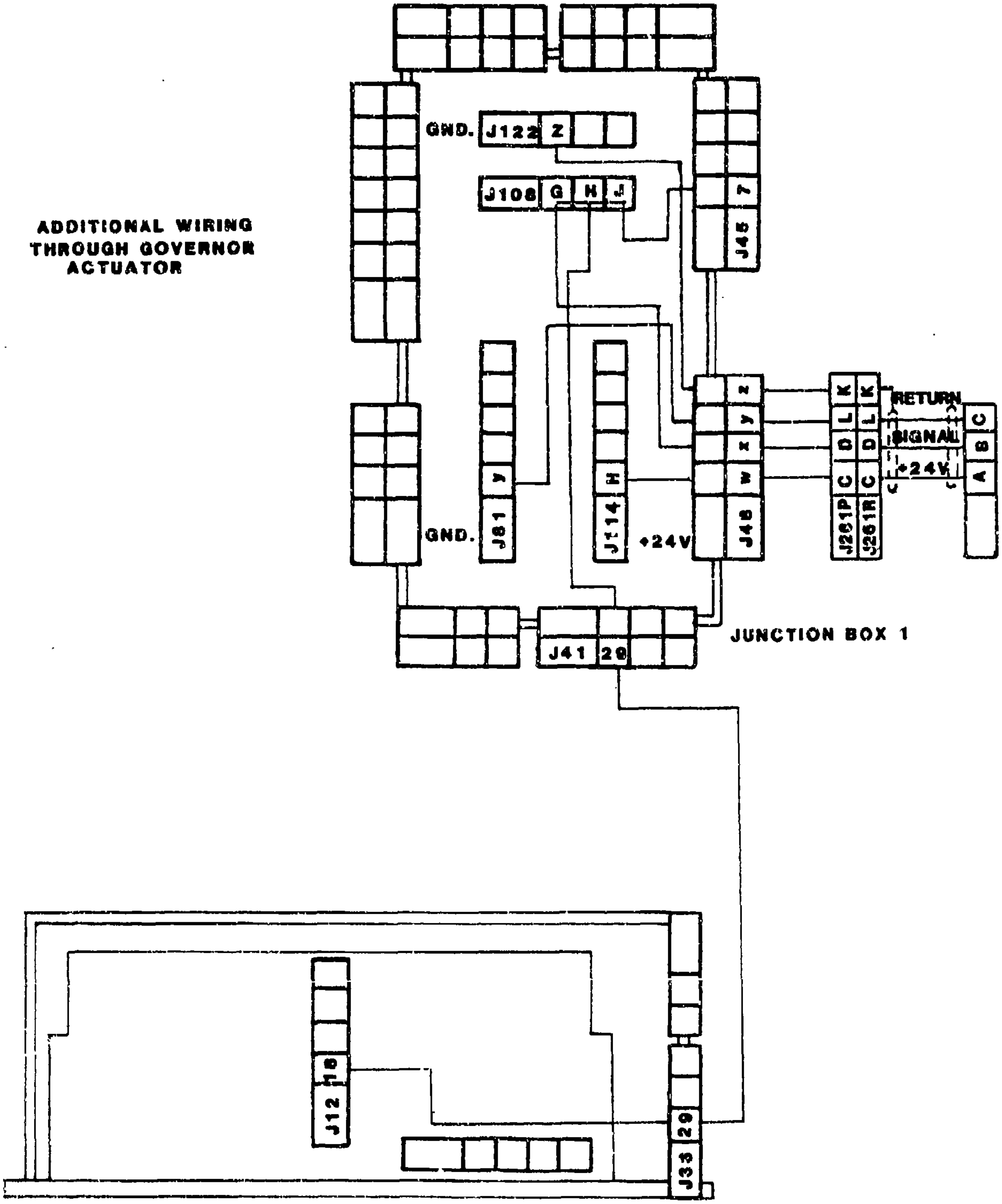
# J12-P16 INPUT PORT WATERJET SPEED



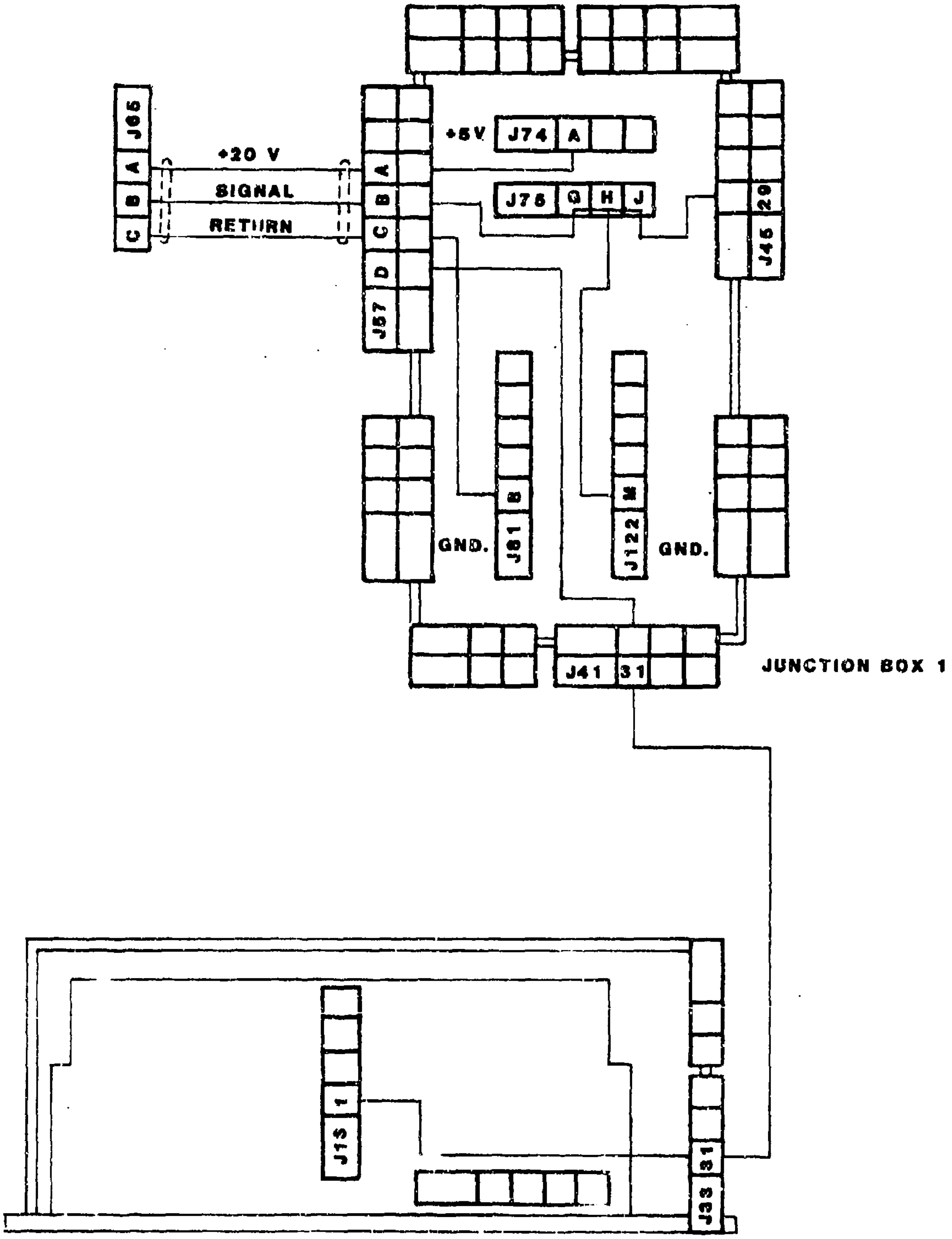
# J12-P17 INPUT STARBOARD WATERJET SPEED



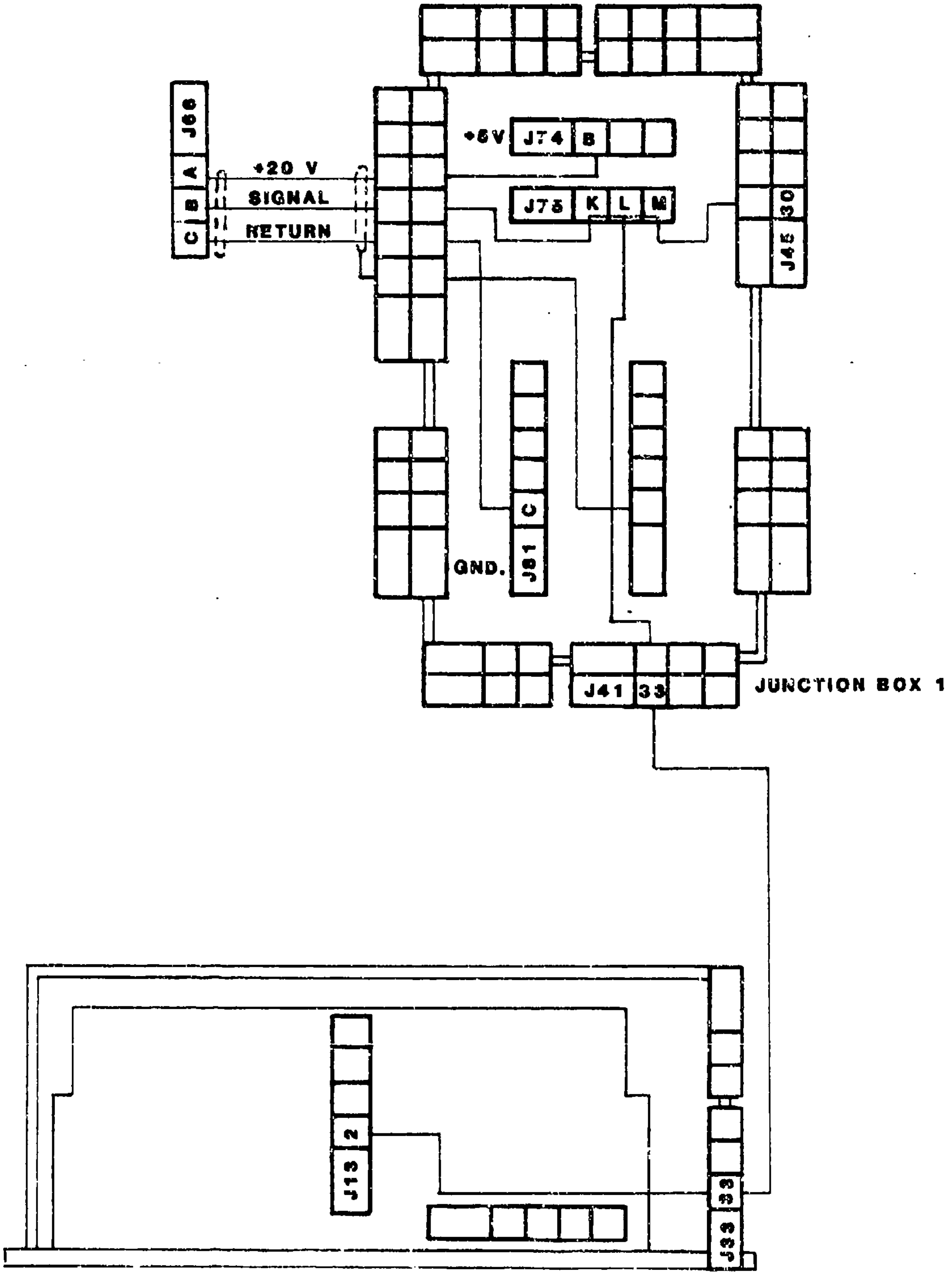
# J12-P18 INPUT ENGINE SPEED



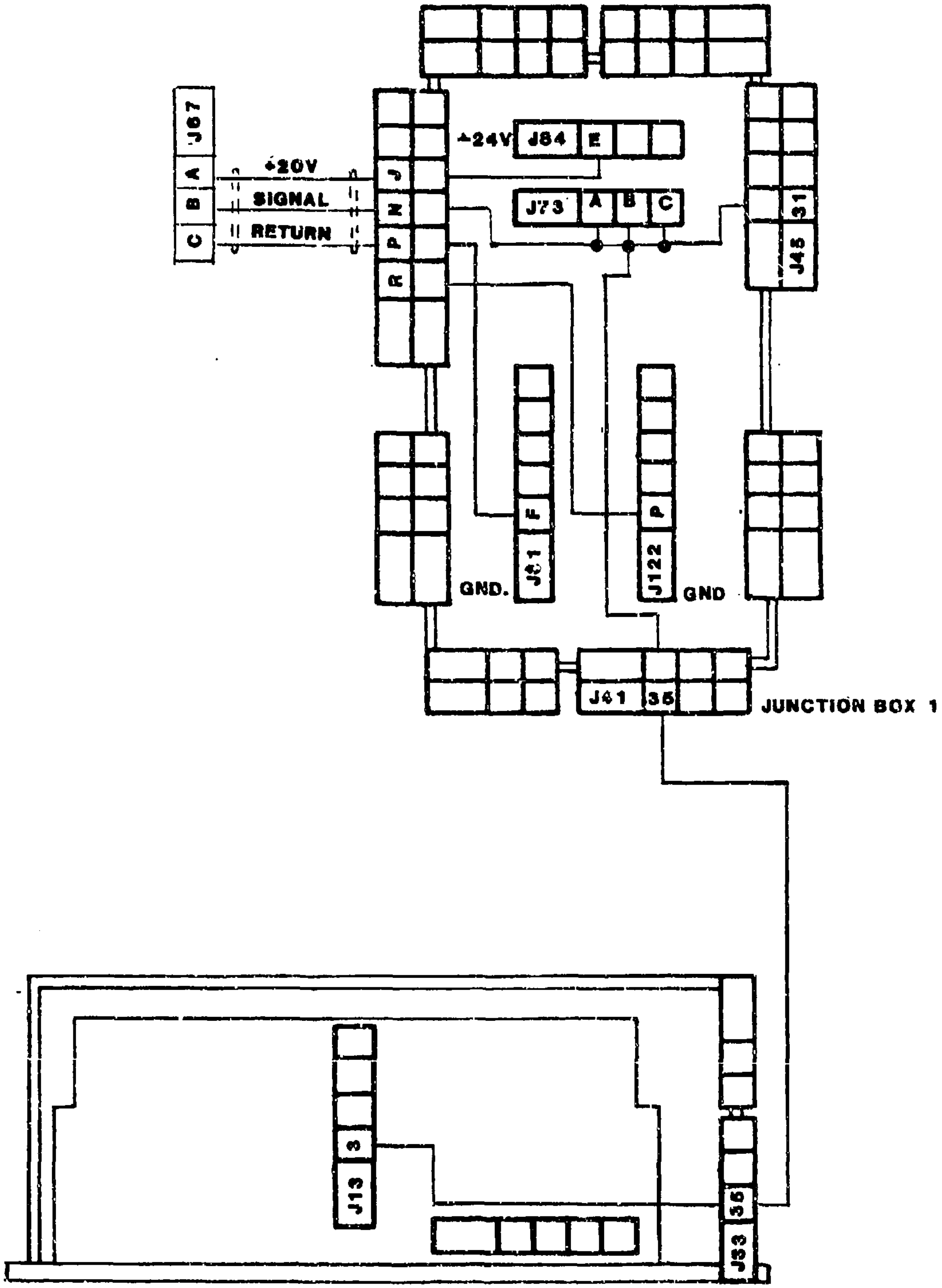
# J13-P1 INPUT DESIRED HIGH MOTOR SPEED



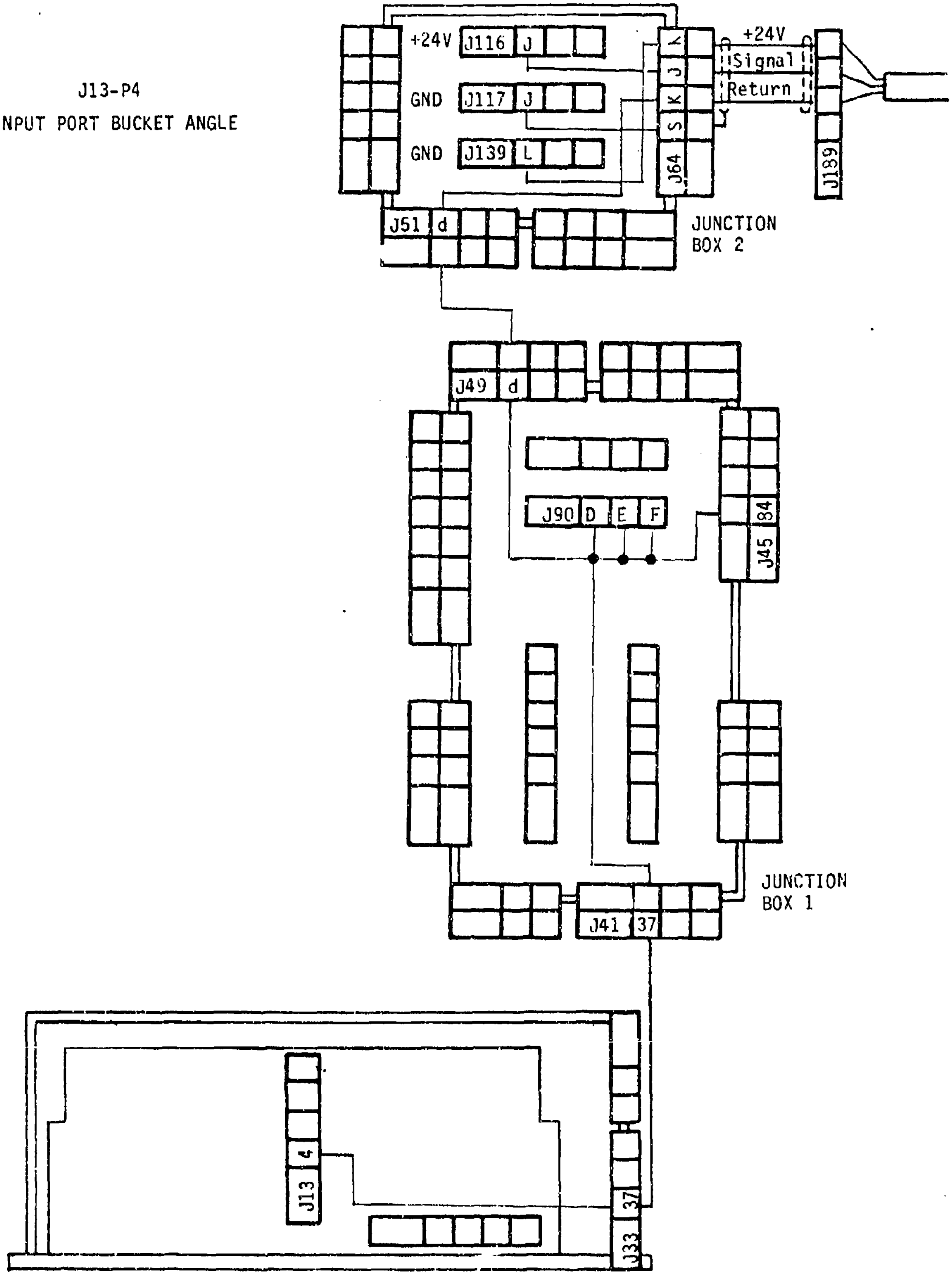
# J13-P2 INPUT DESIRED HIGH MOTOR SPEED RATIO



# J13-P3 INPUT DESIRED TURN RATIO

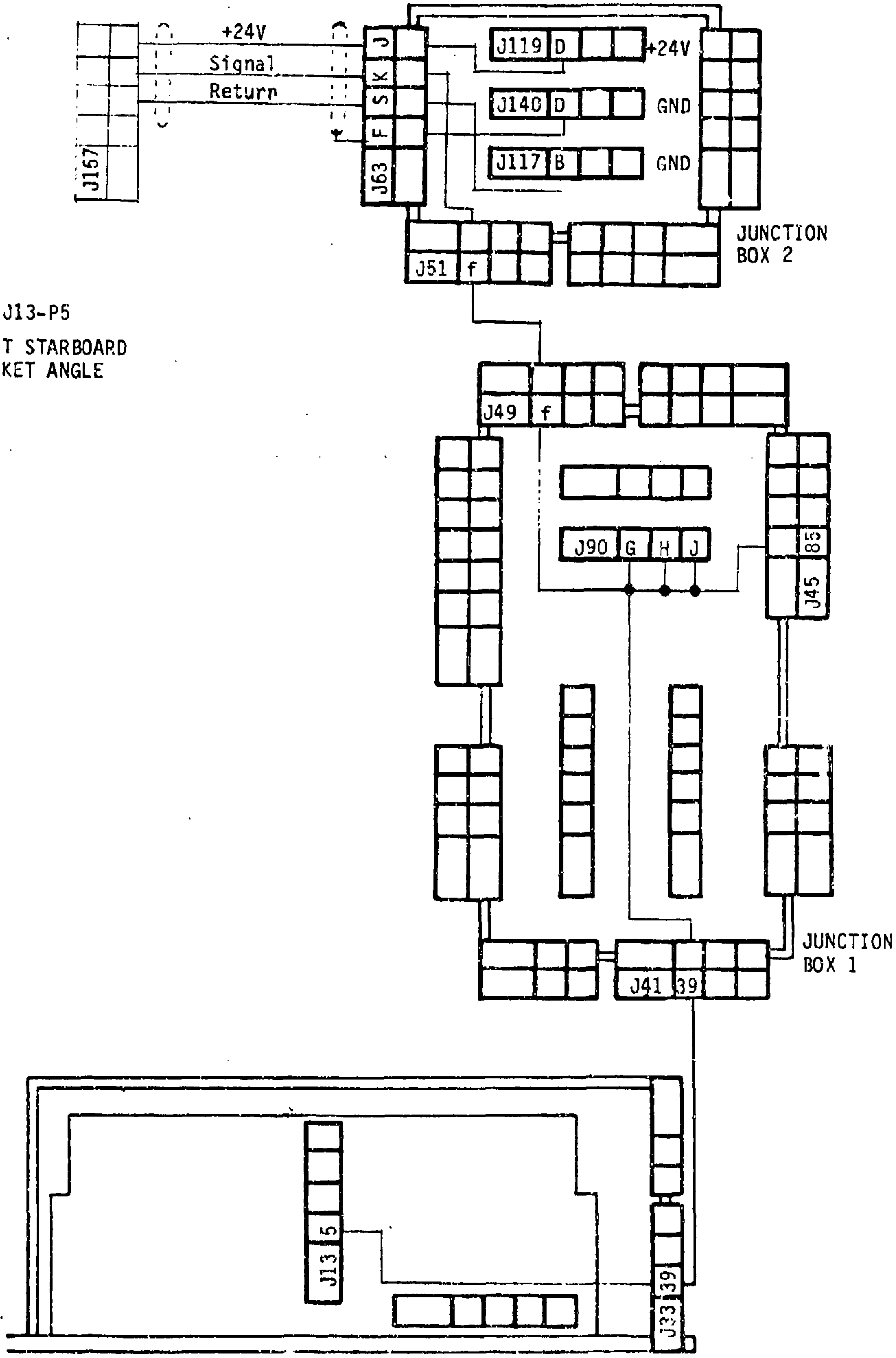


J13-P4
INPUT PORT BUCKET ANGLE

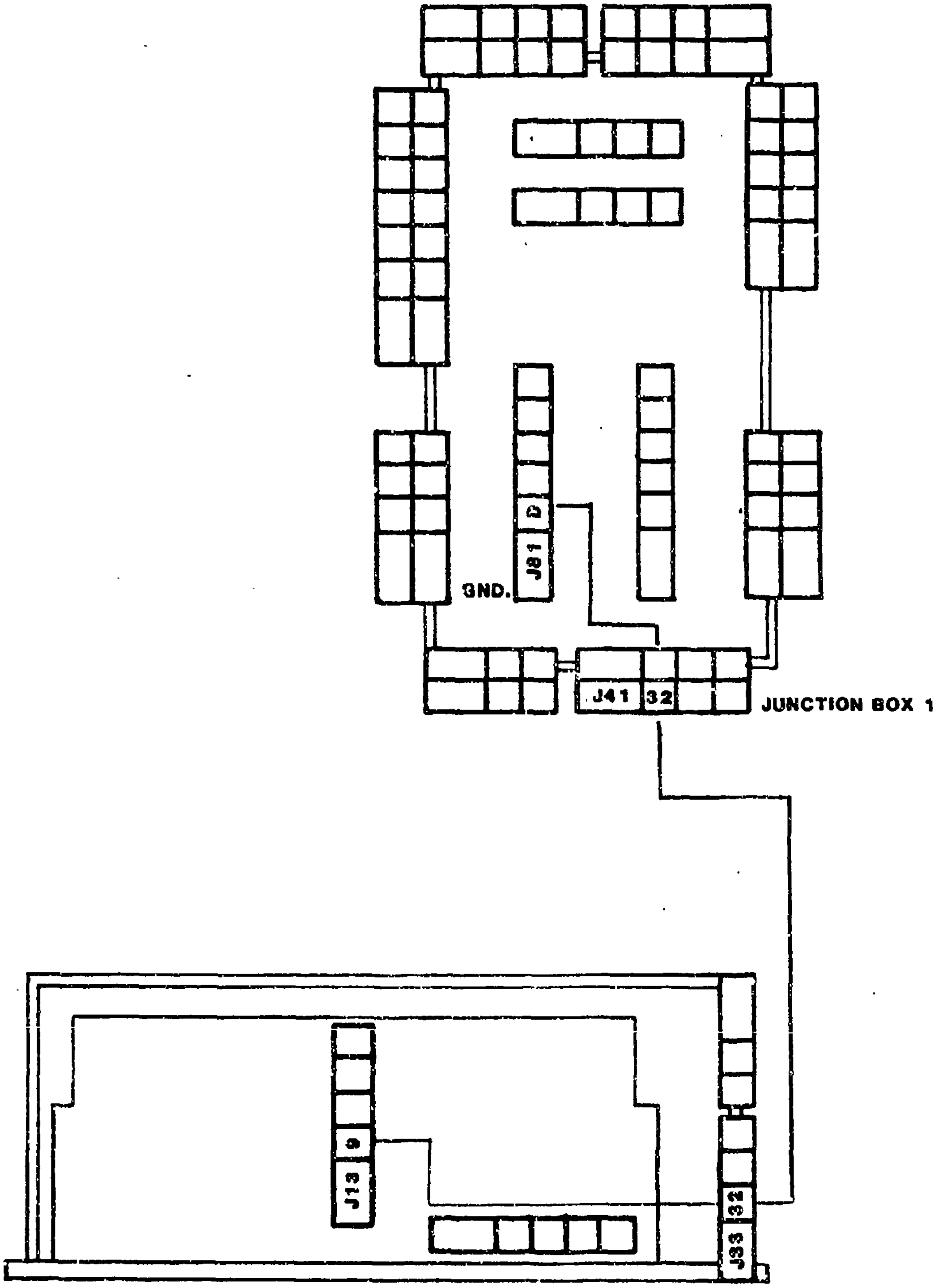


J13-P5

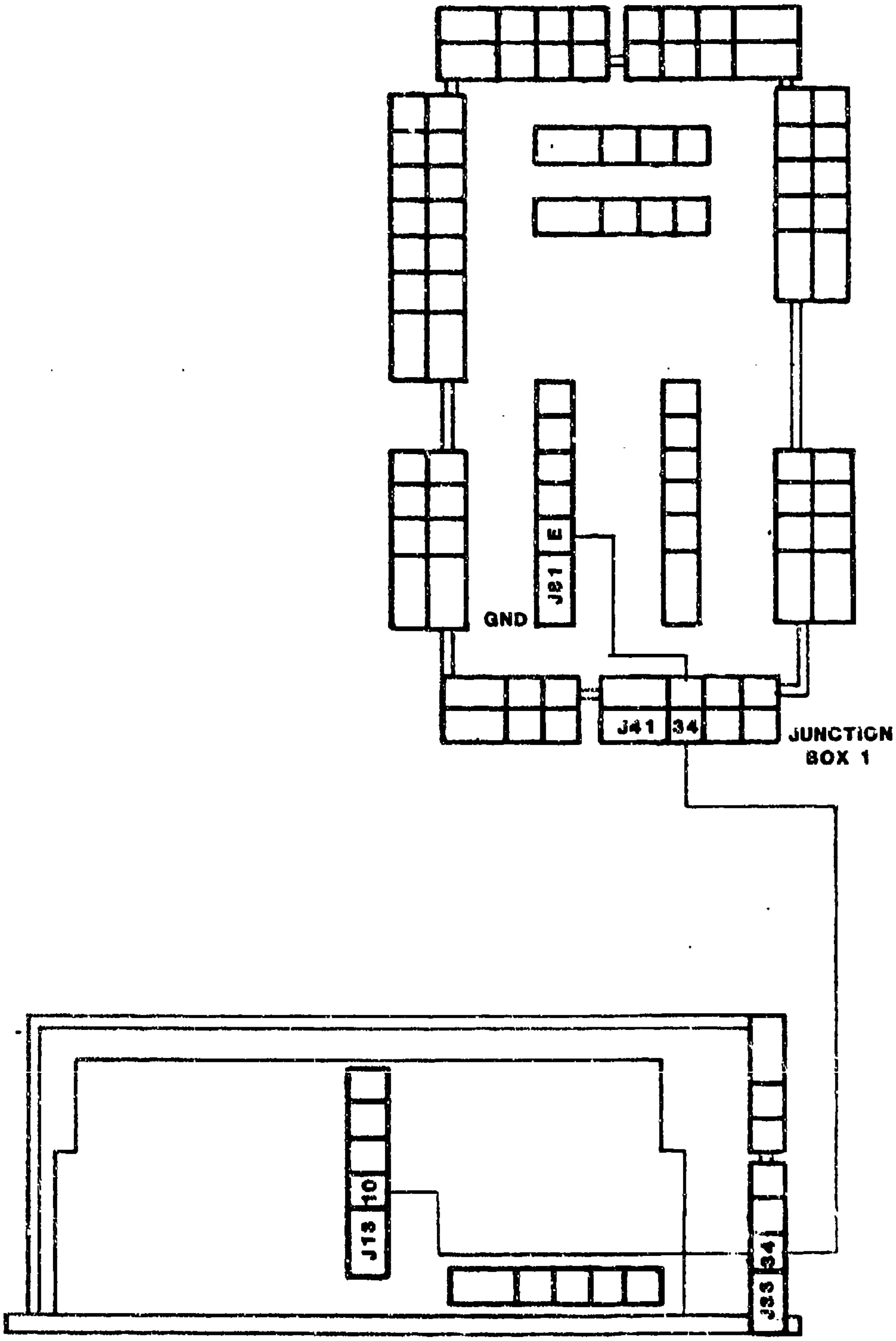
INPUT STARBOARD BUCKET ANGLE



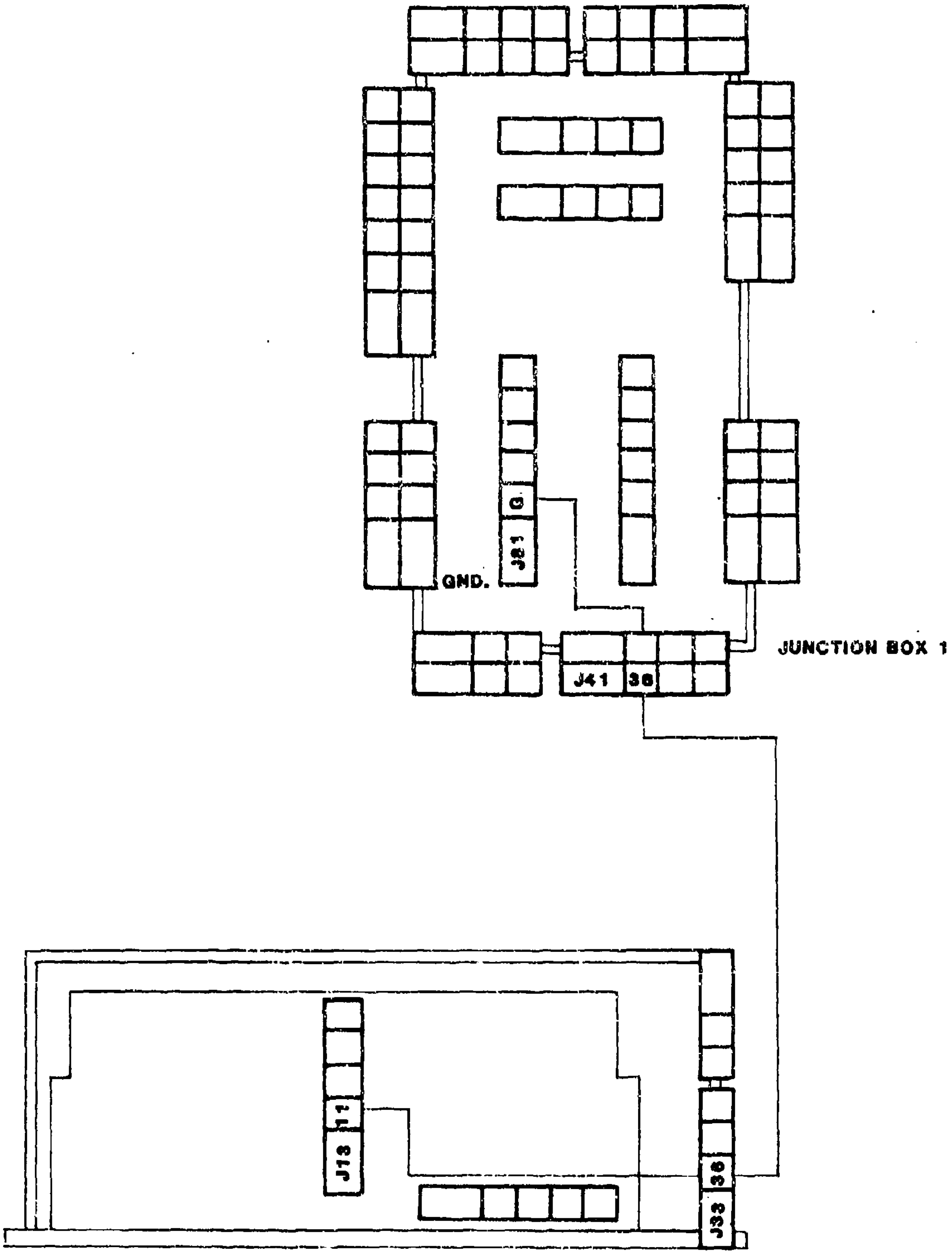
## J13-P9 GROUND



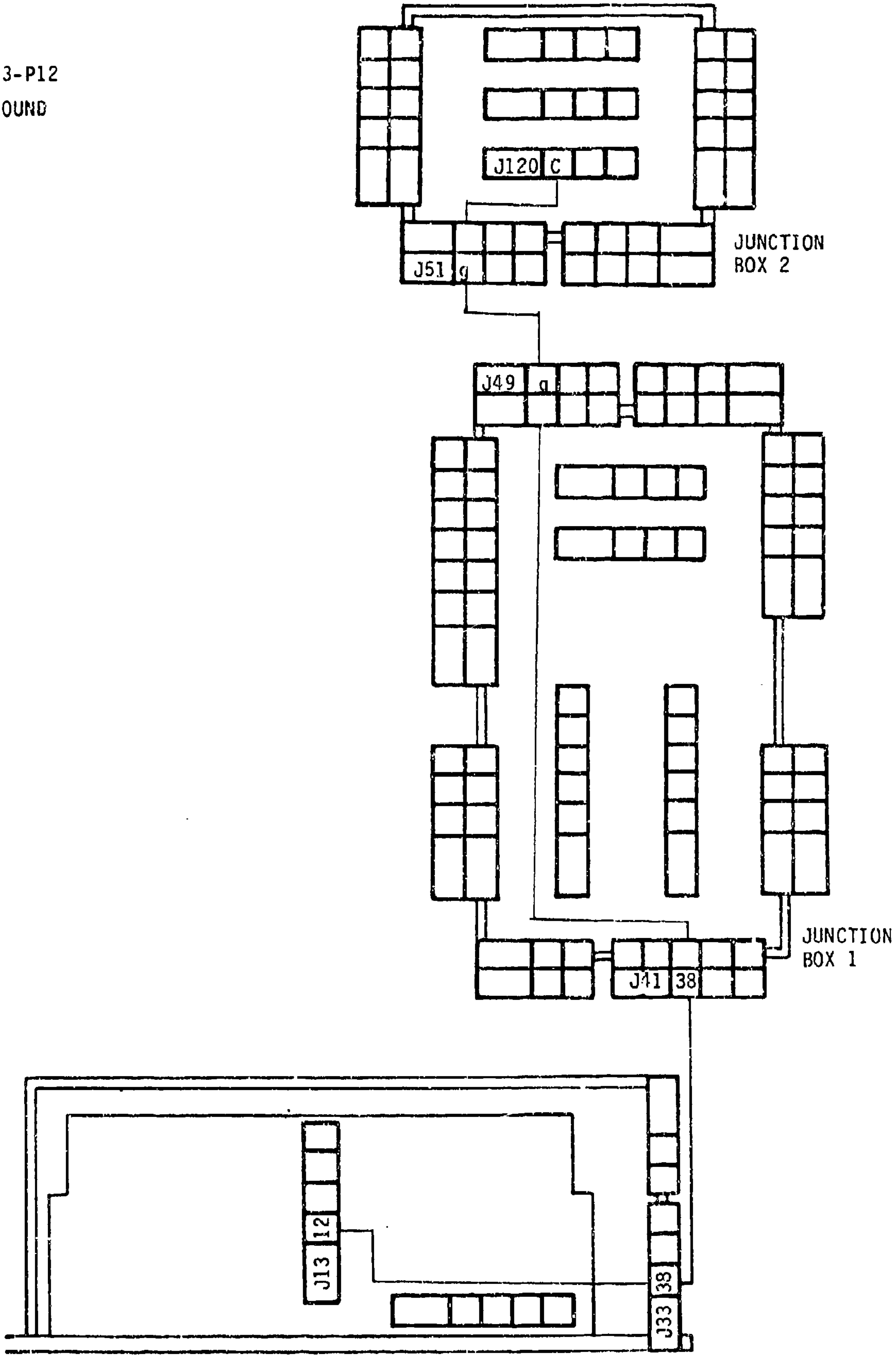
# J13-P10 GROUND



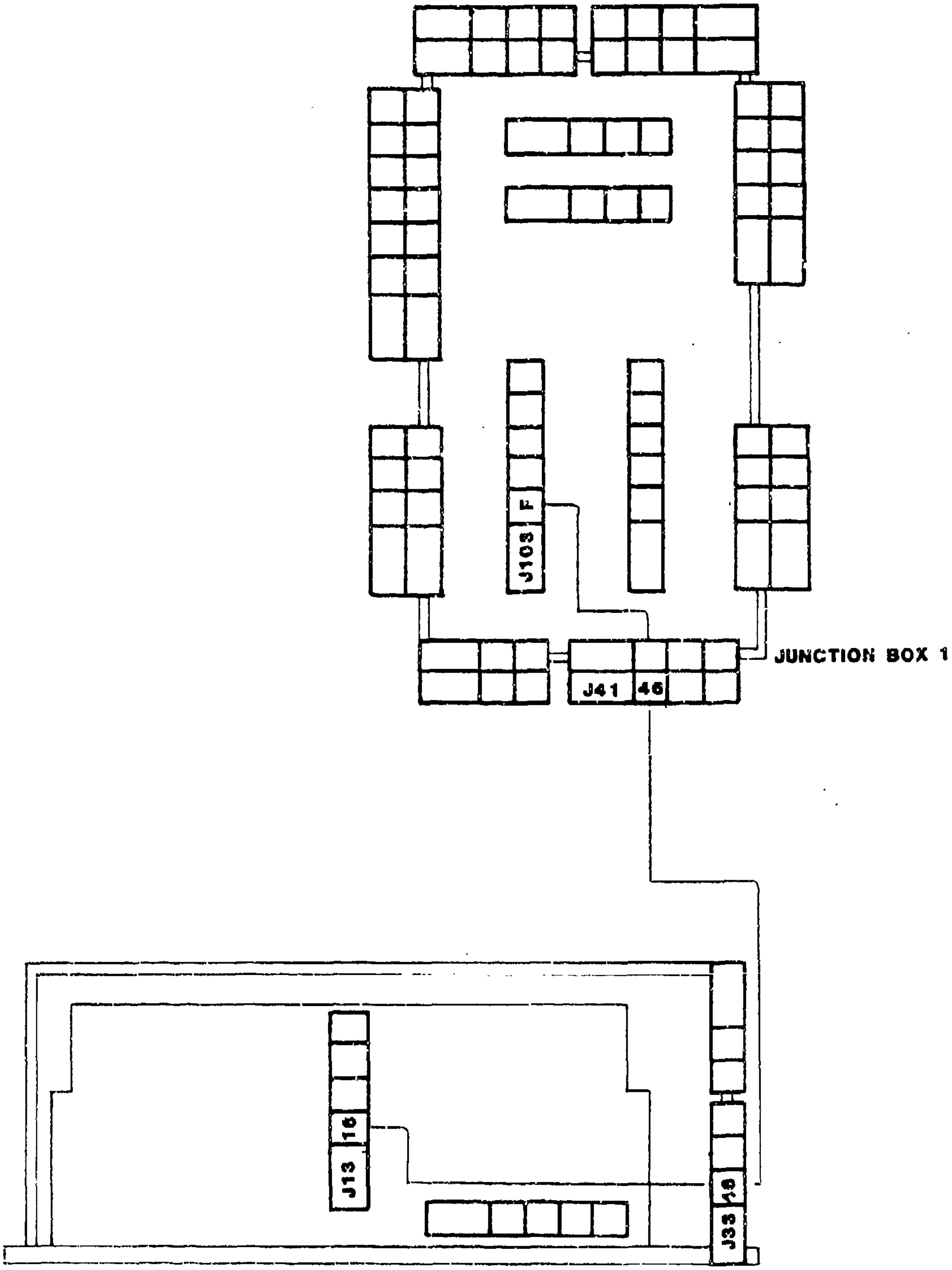
# J13-P11 GROUND



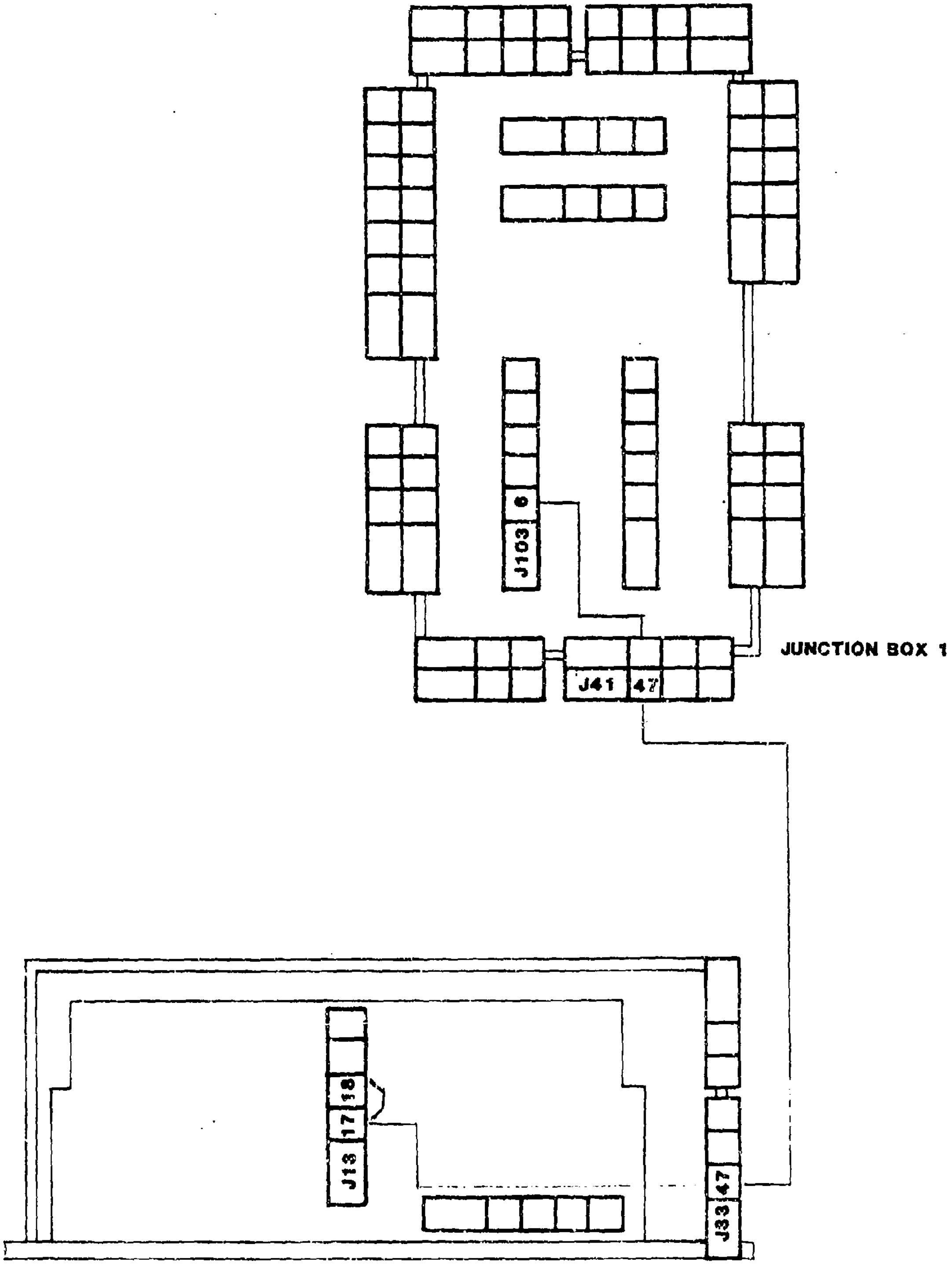
J13-P12
GROUND



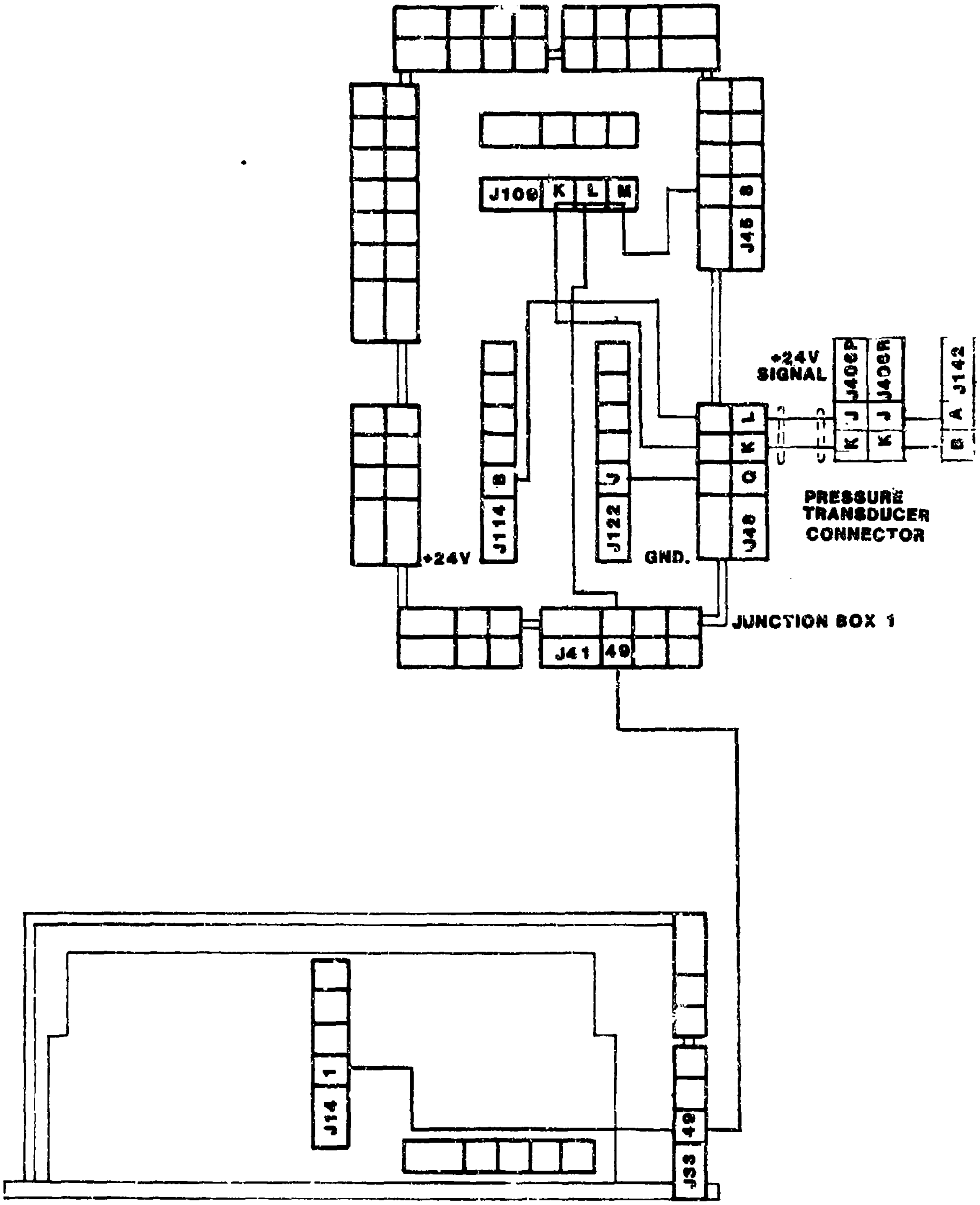
# J13-P16 GROUND



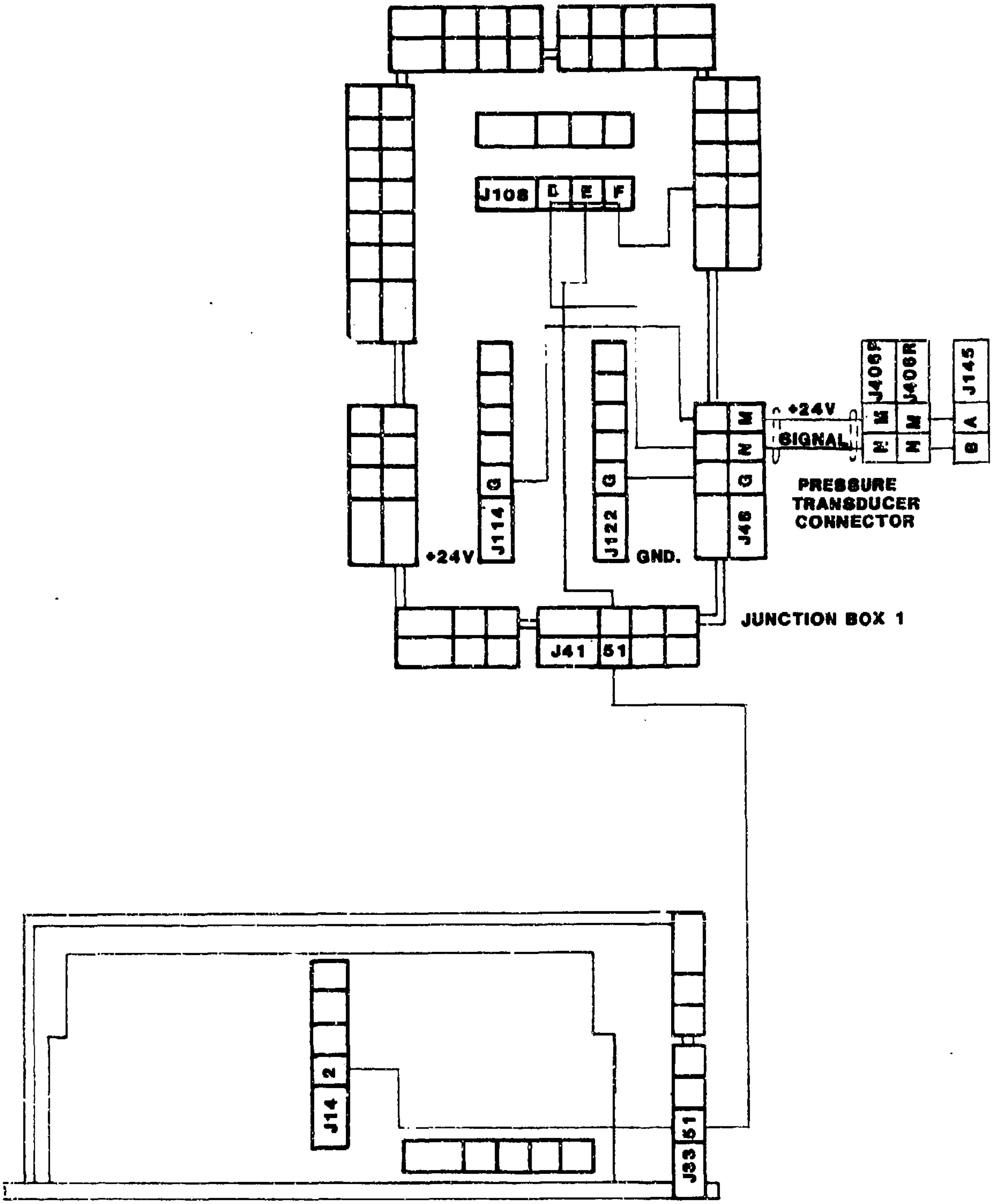
# J13-P17,P18 COMPUTER GROUND



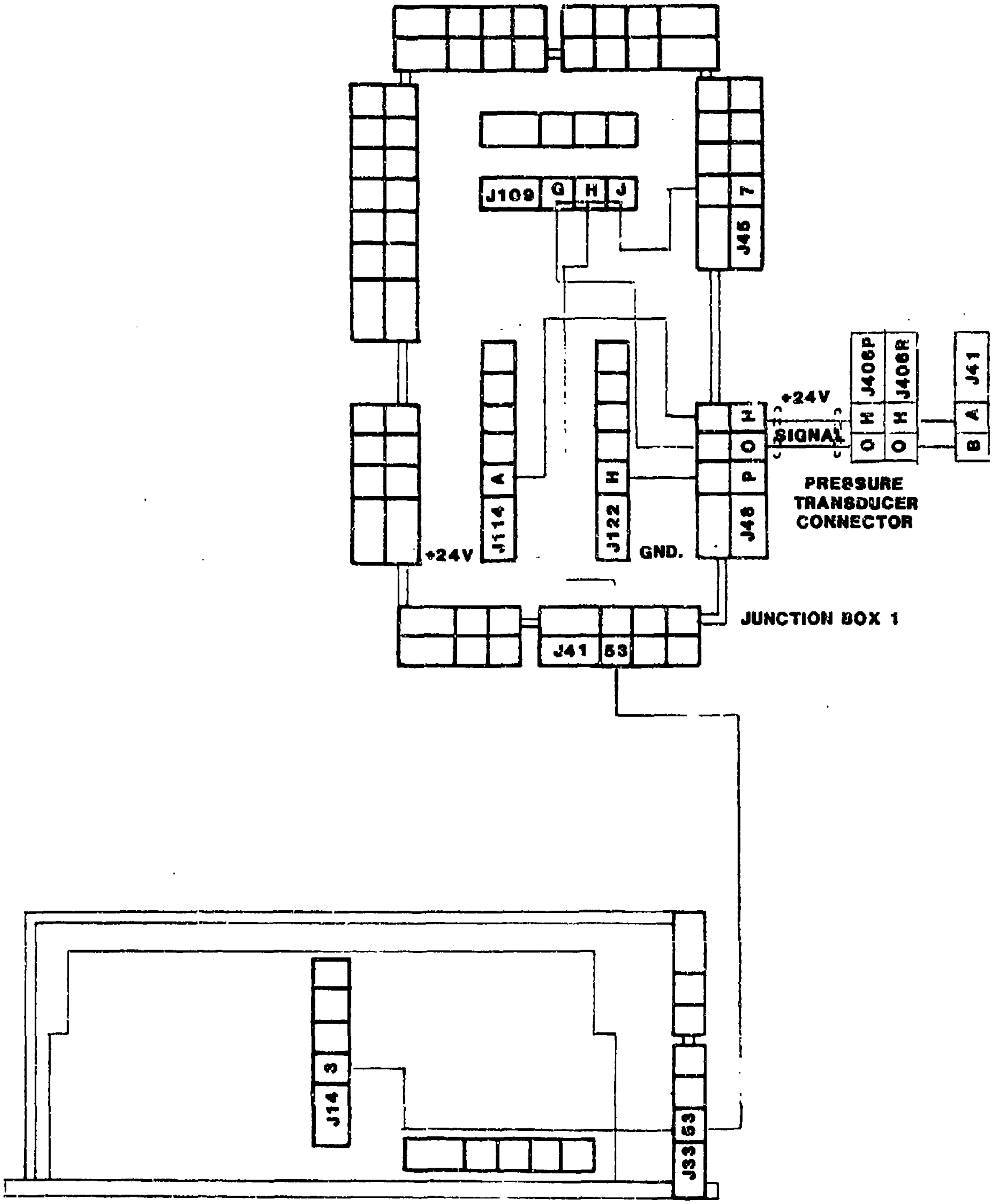
## J14-P1 INPUT PORT FORWARD MOTOR PRESSURE



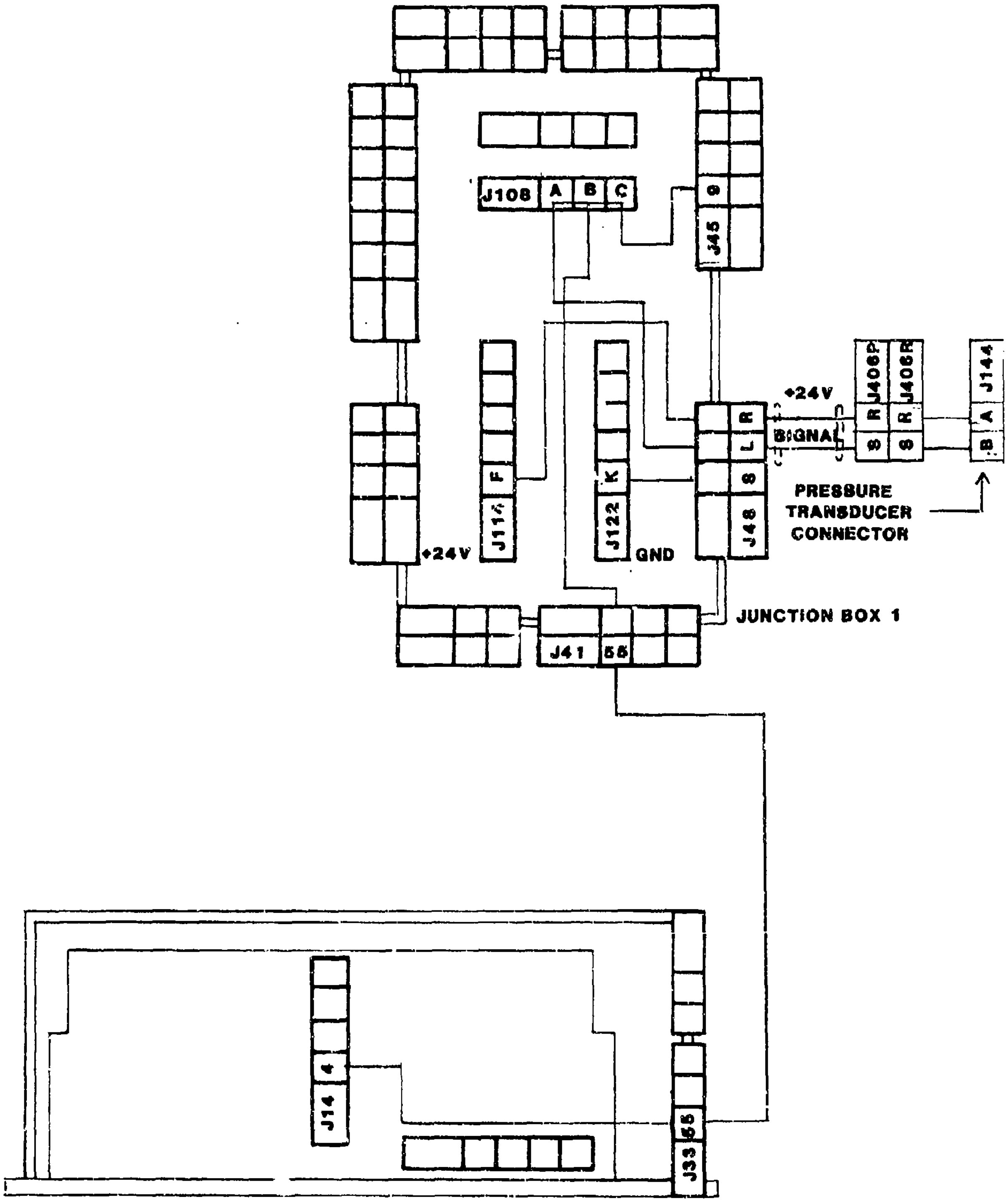
# J14-P2 INPUT STARBOARD FORWARD MOTOR PRESSURE

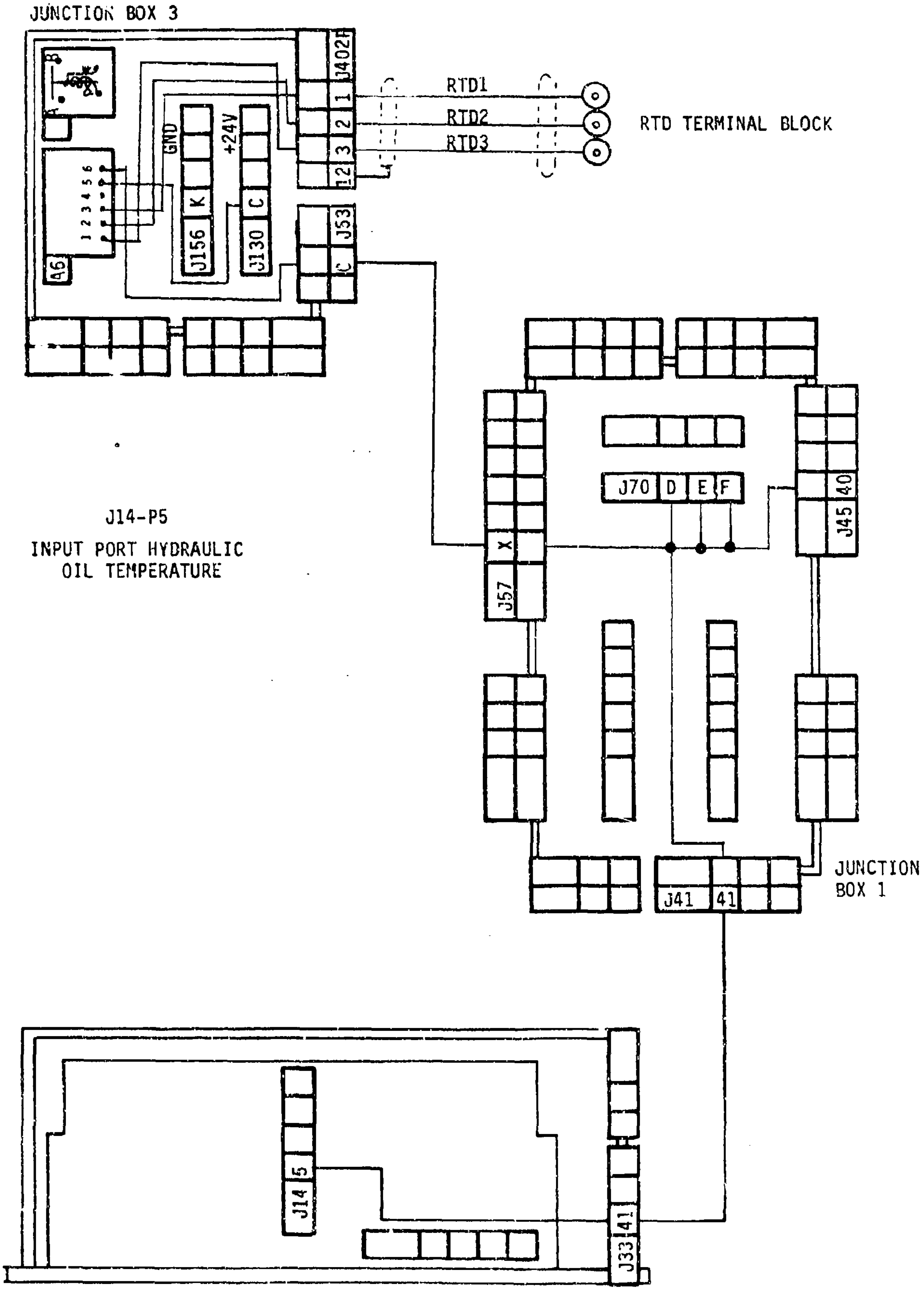


# J14-P3 INPUT PORT AFT MOTOR PRESSURE



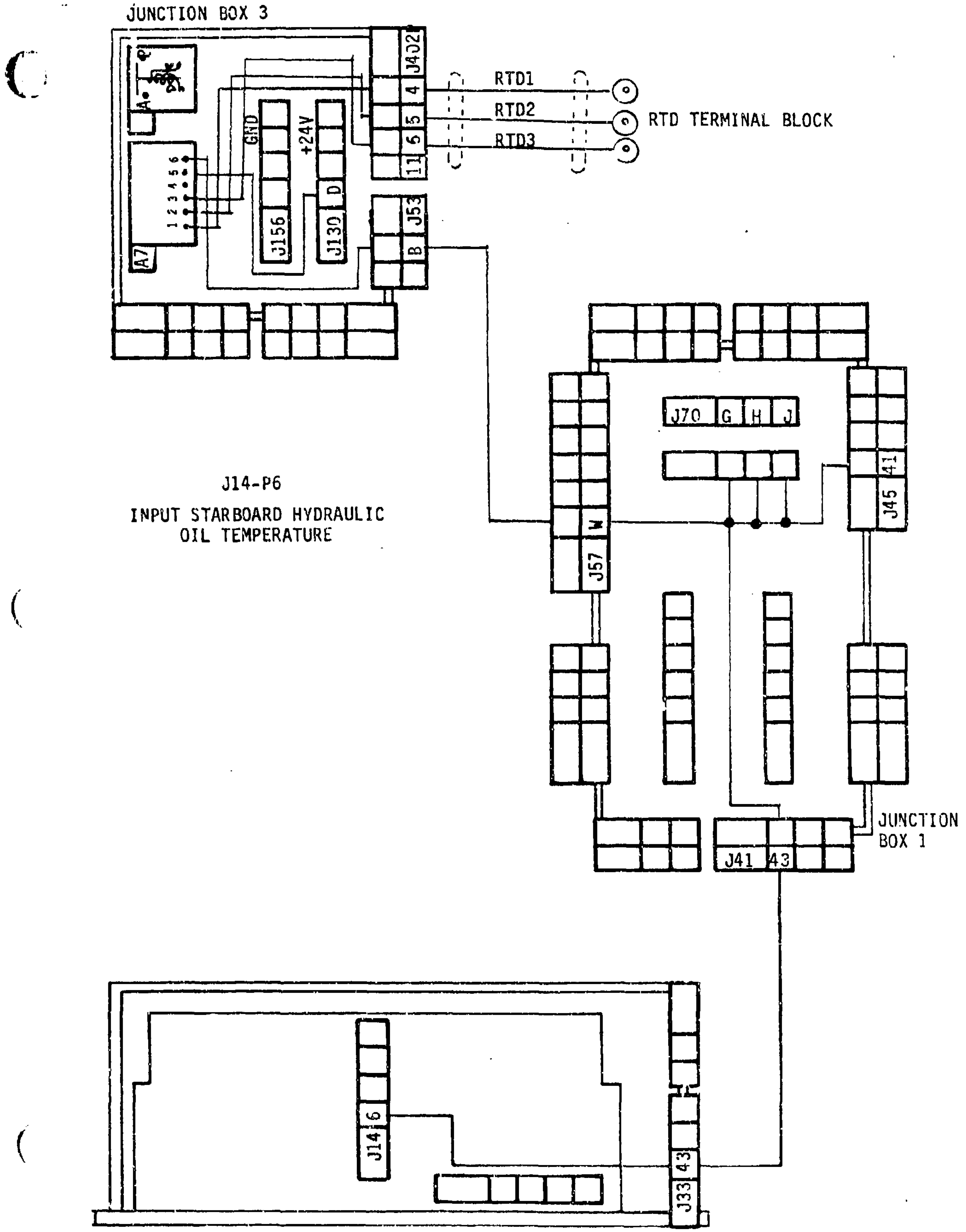
# J14-P4 INPUT STARBOARD AFT MOTOR PESSURE





J14-P5

INPUT PORT HYDRAULIC
OIL TEMPERATURE

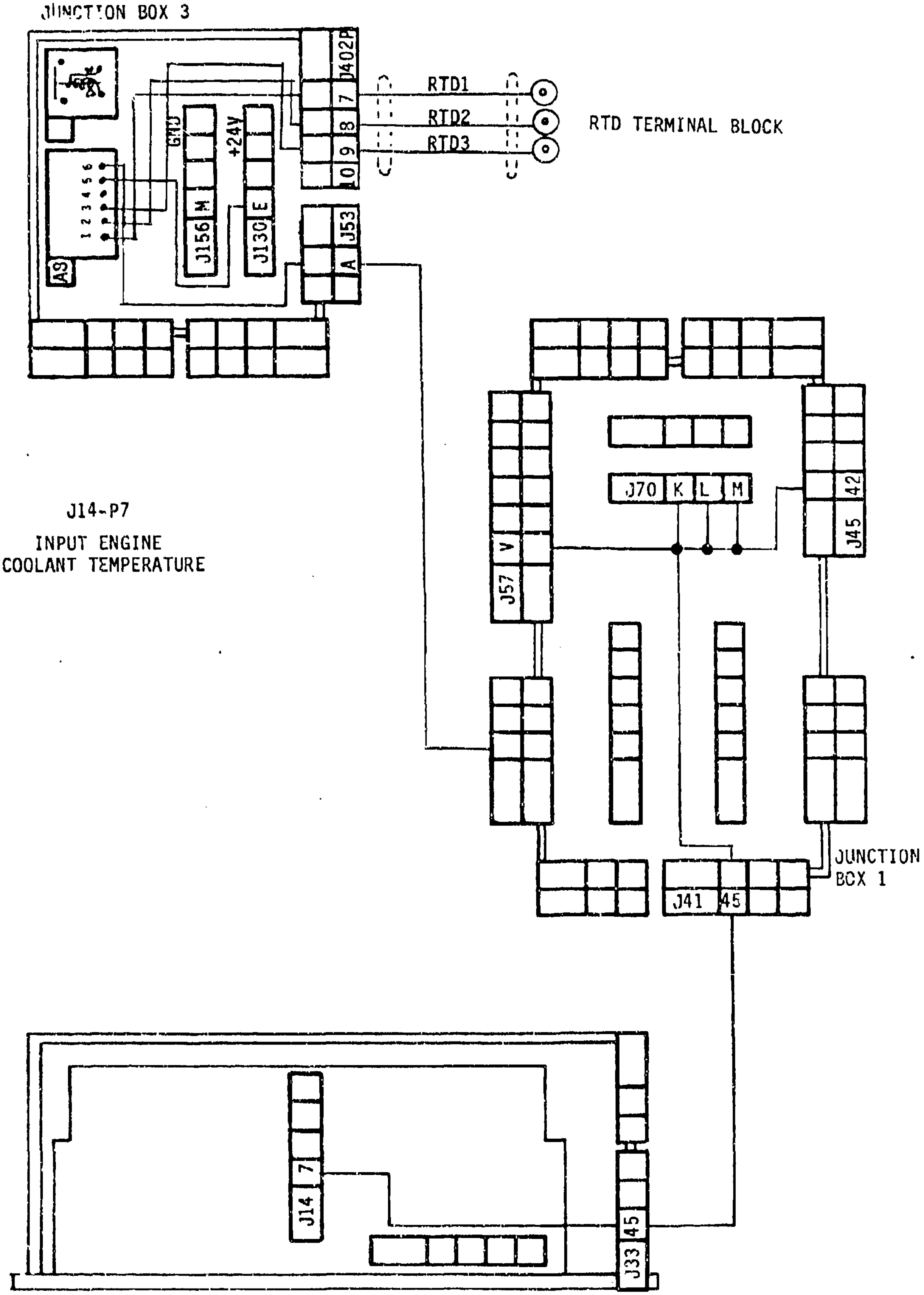


J14-P6

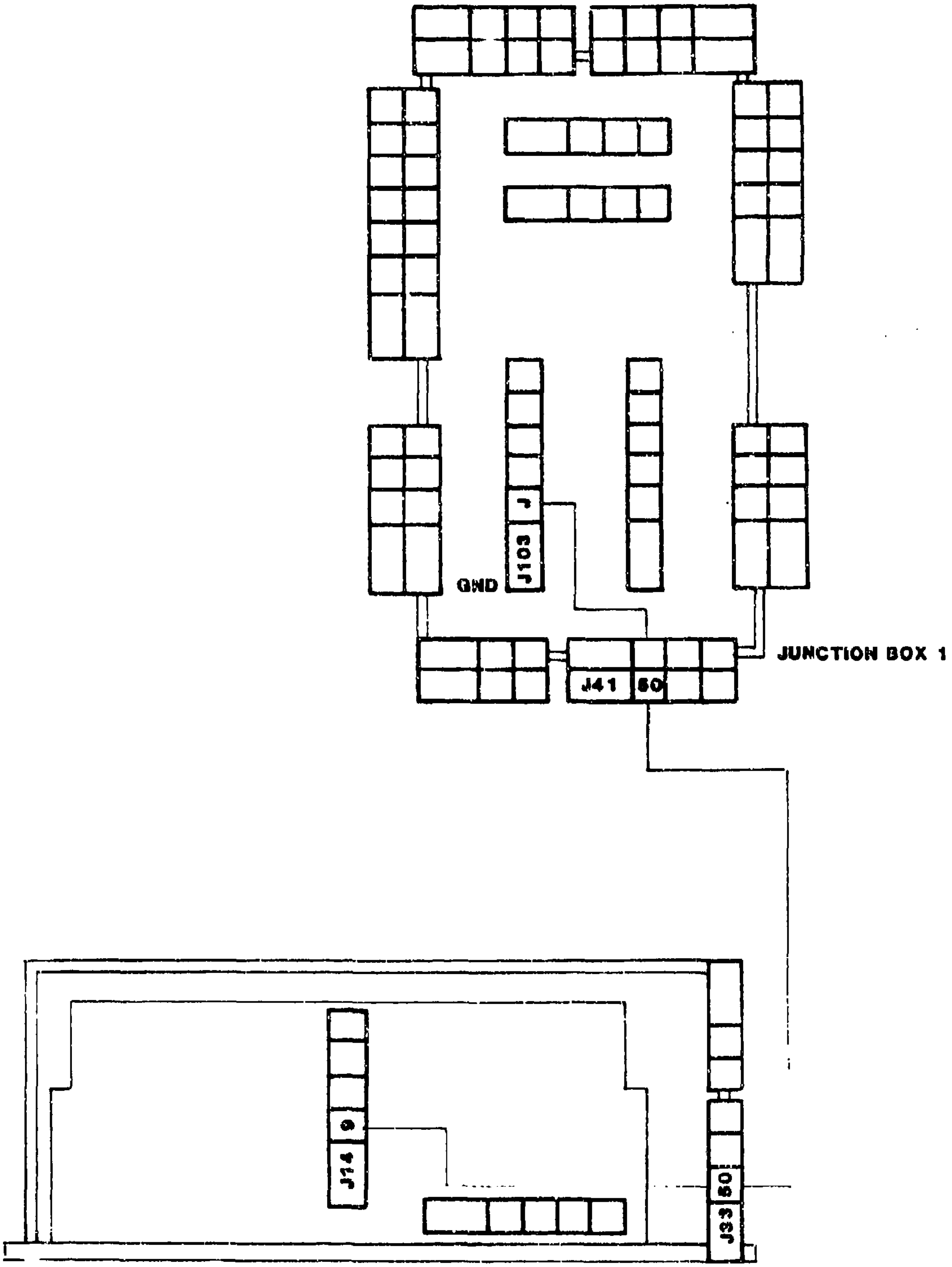
INPUT STARBOARD HYDRAULIC
OIL TEMPERATURE

J14-P7

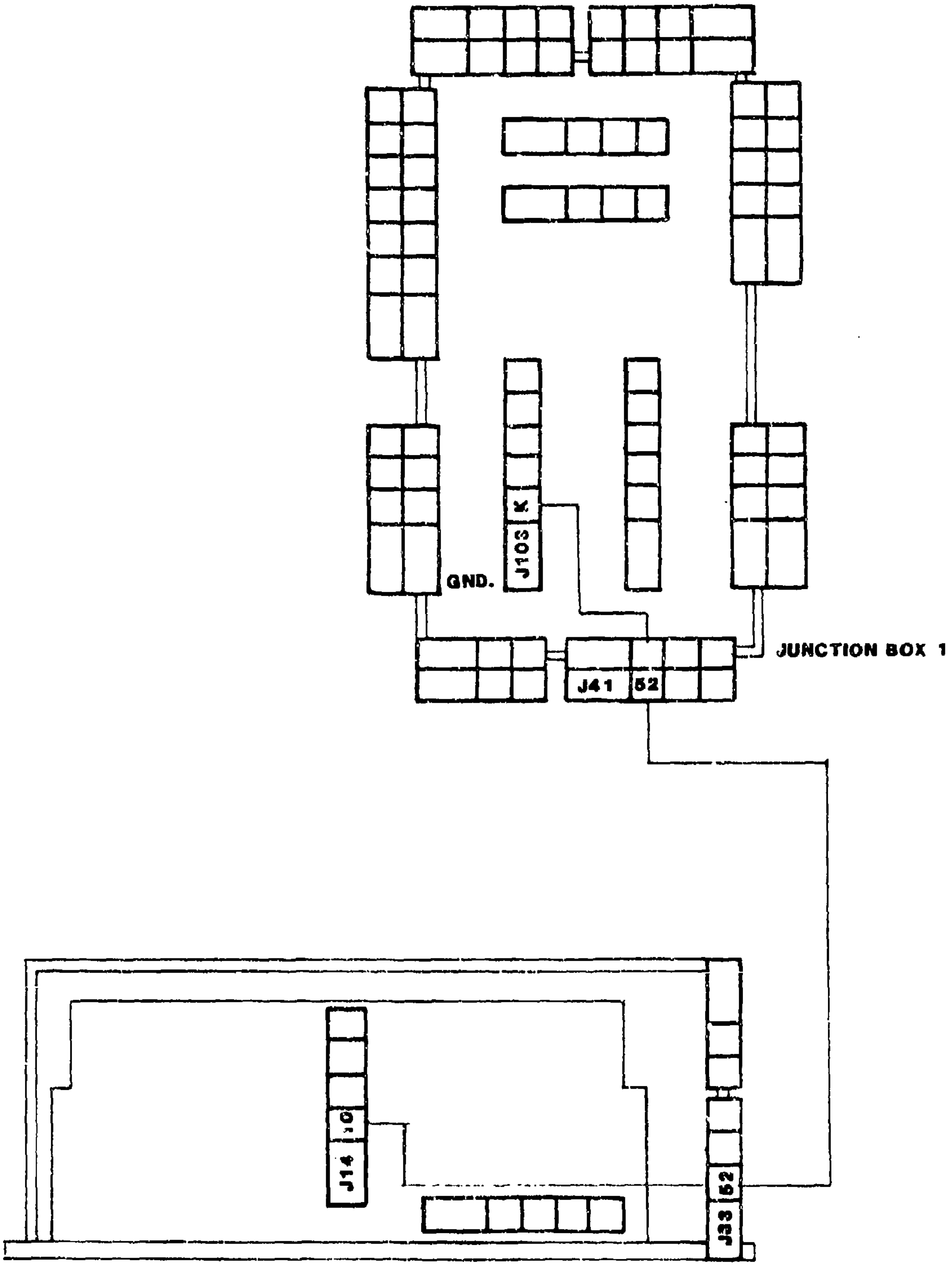
INPUT ENGINE COOLANT TEMPERATURE



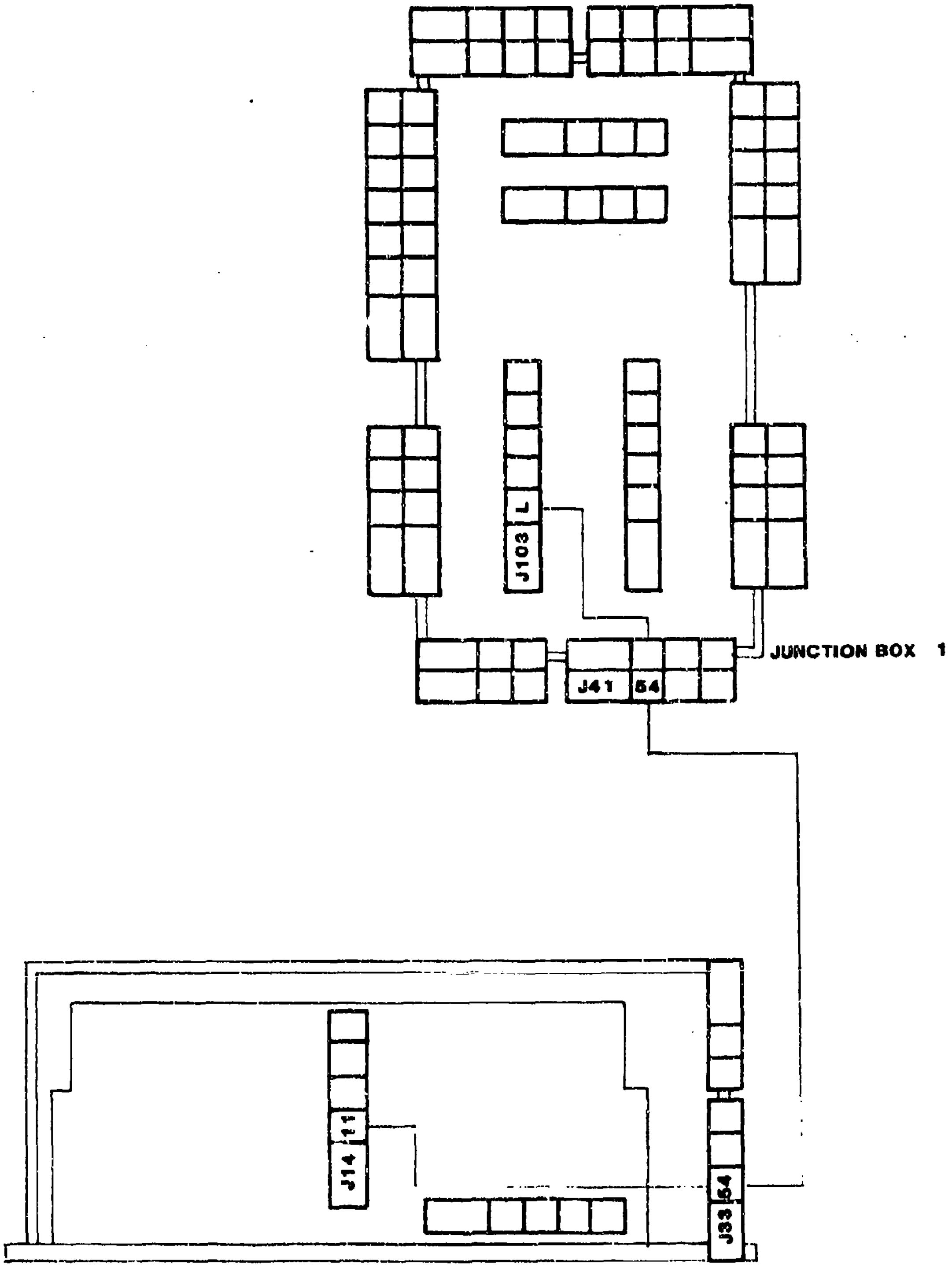
# J14-P9 GROUND



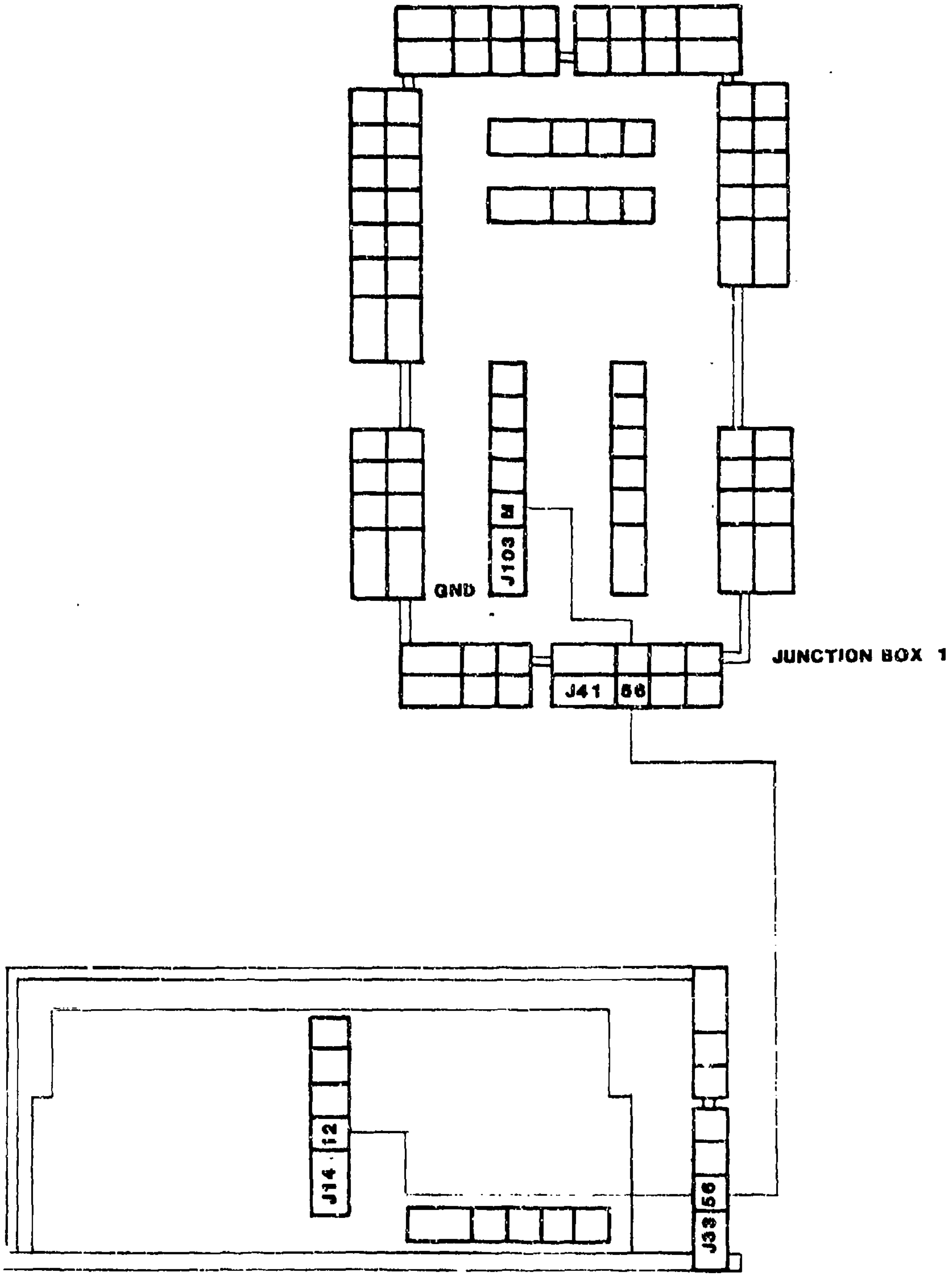
# J14-P10 GROUND



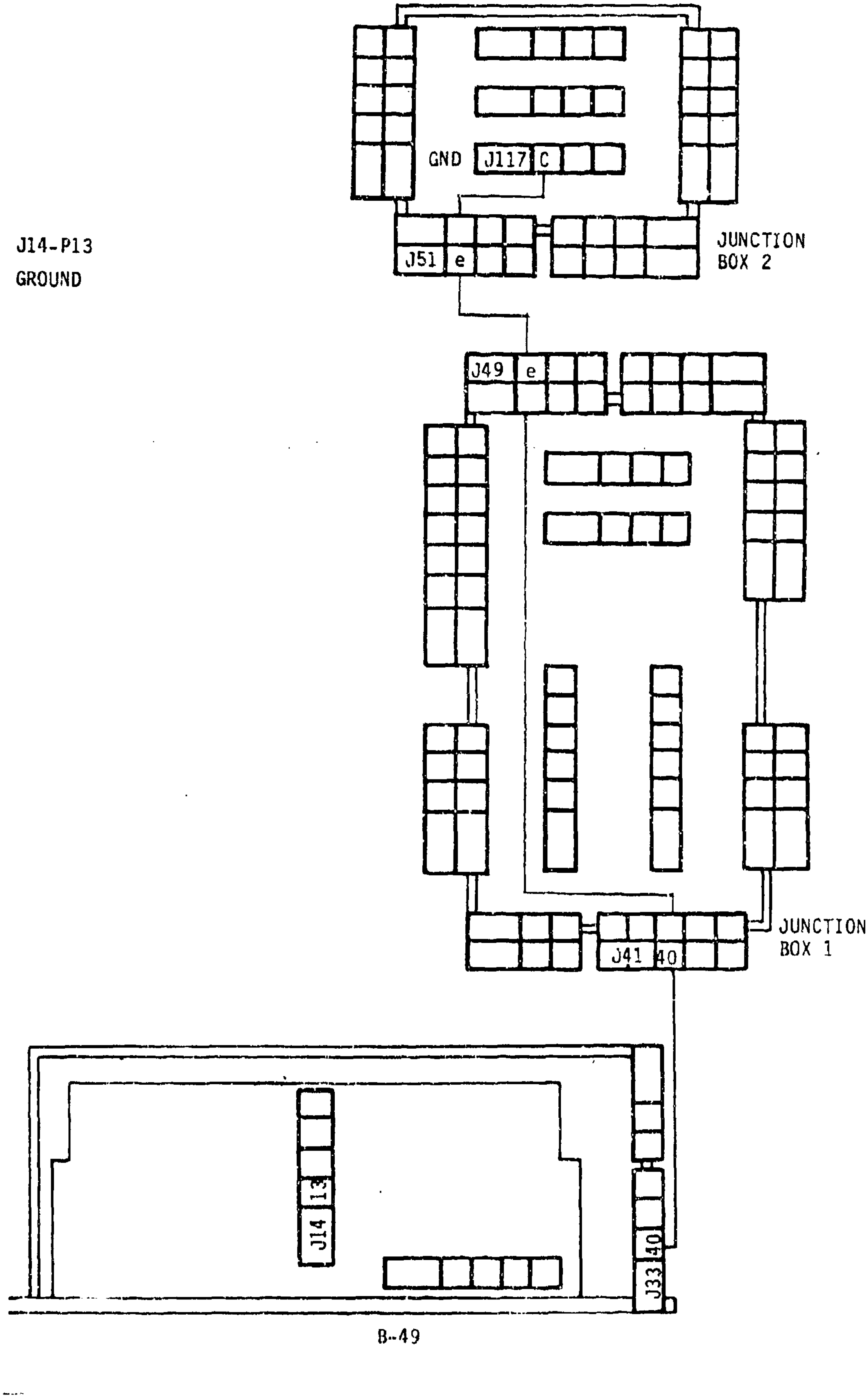
# J14-P11 GROUND



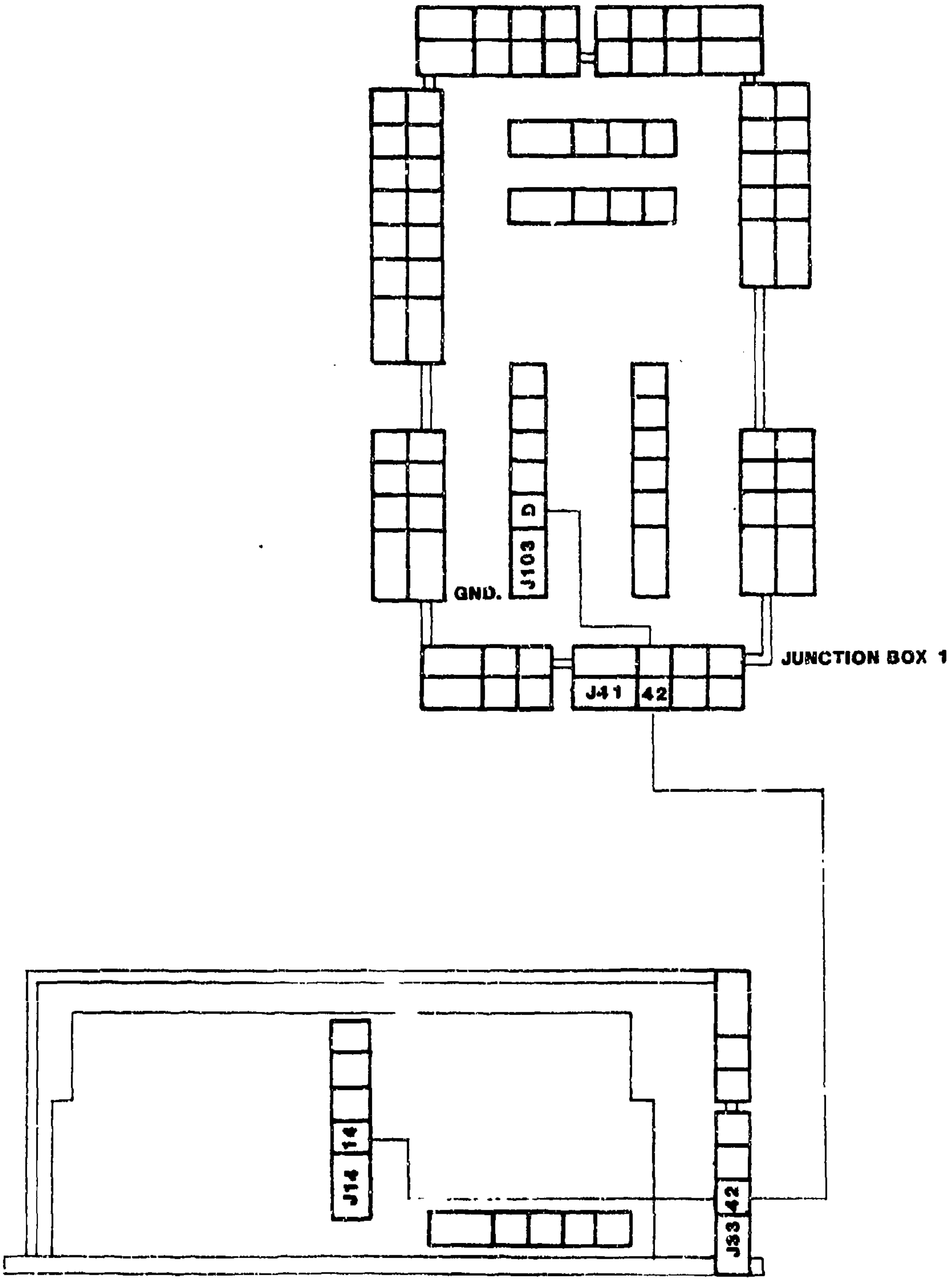
# J14-P12 GROUND



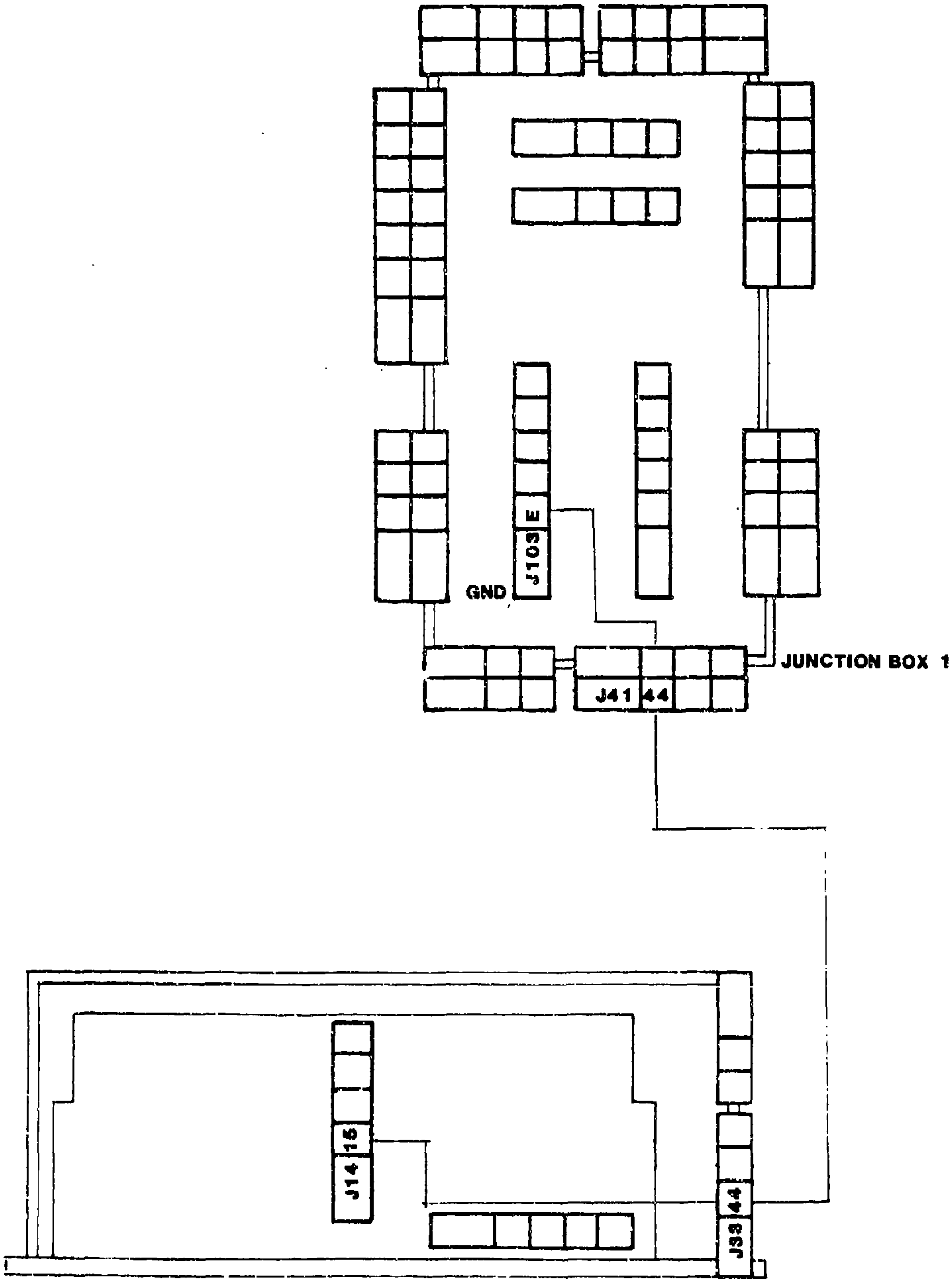
J14-P13
GROUND



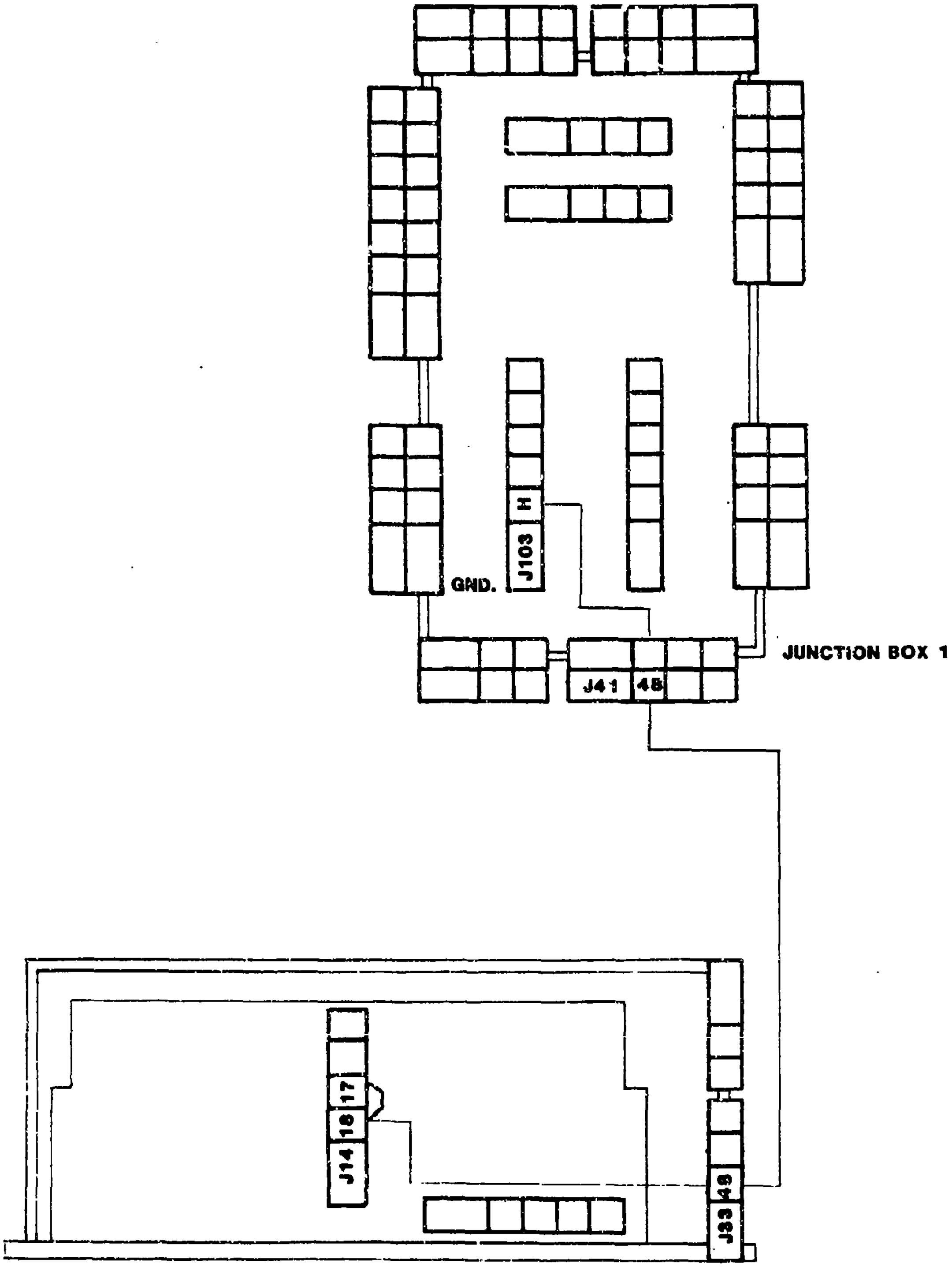
# J14-P14 GROUND



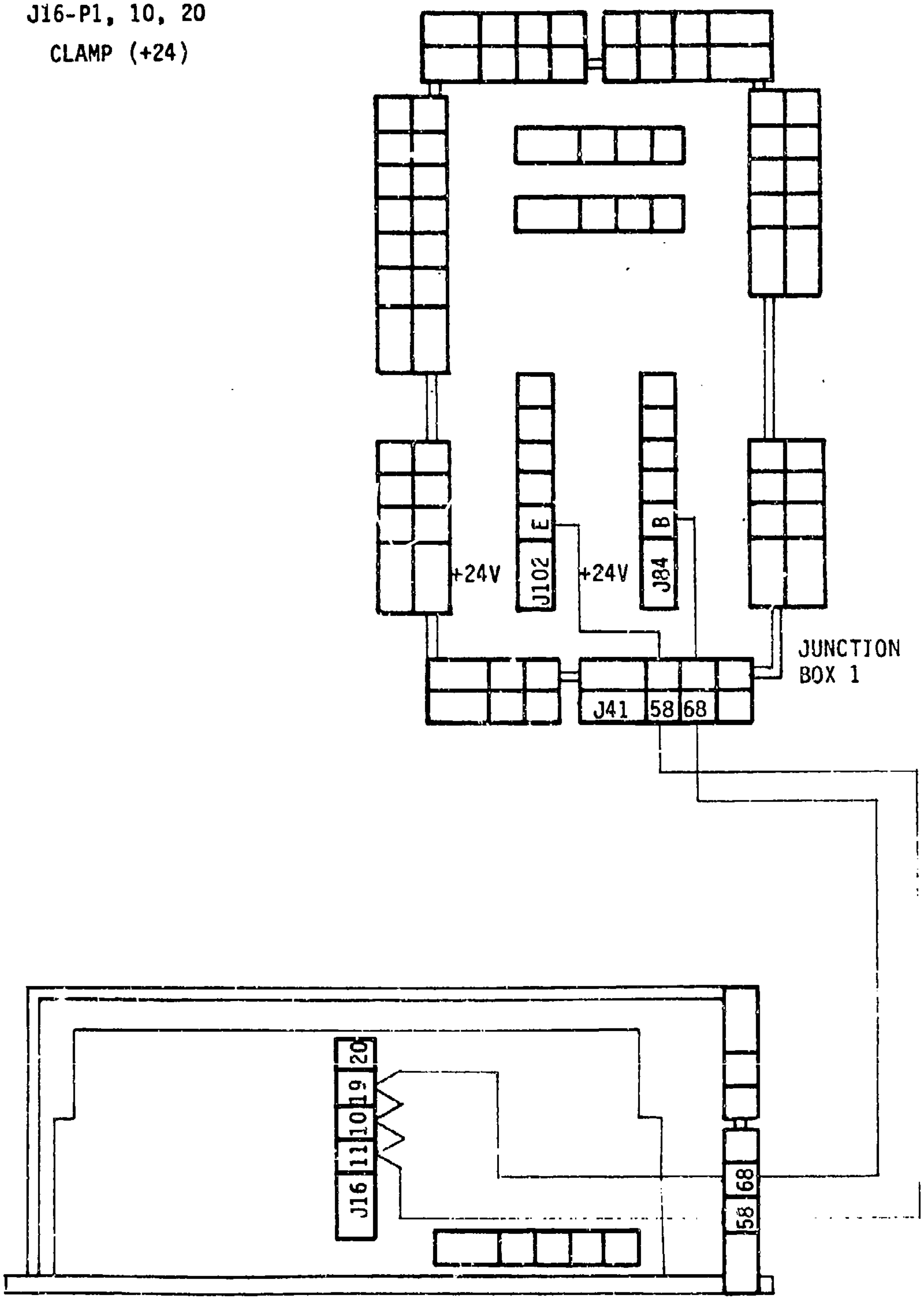
## J14-P15 GROUND

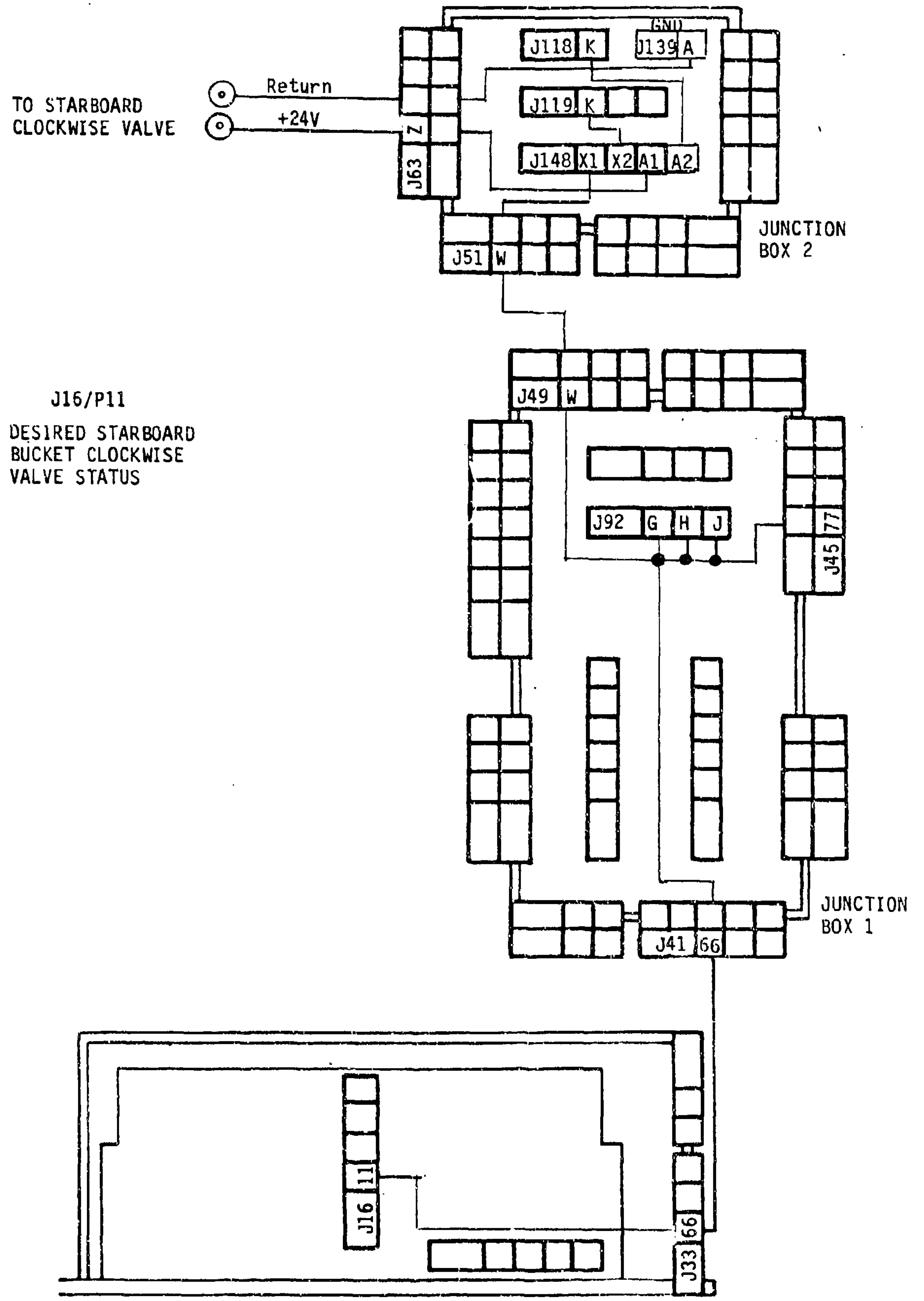


# J14-P18,P17 COMPUTER GROUND



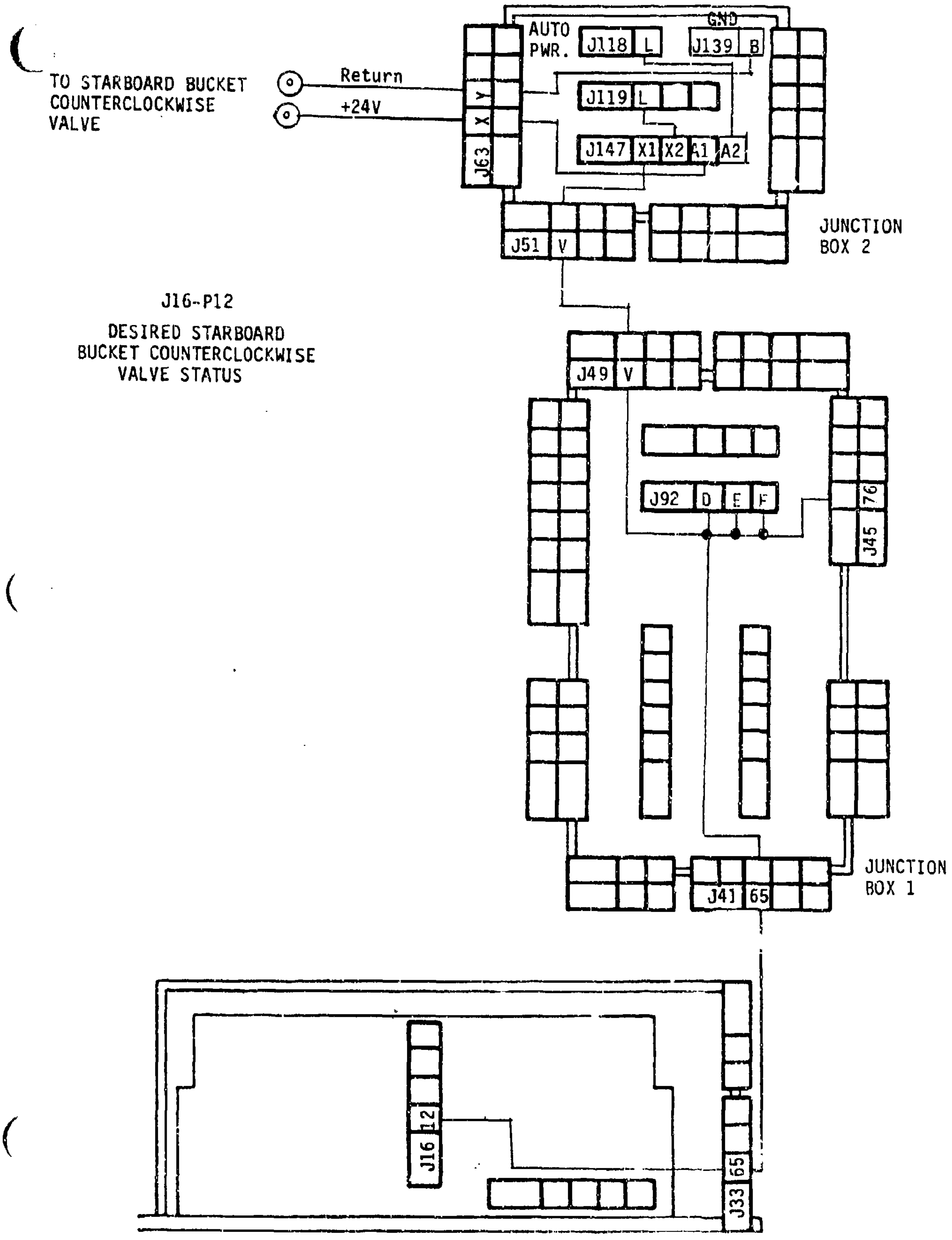
J16-P1, 10, 20
CLAMP (+24)

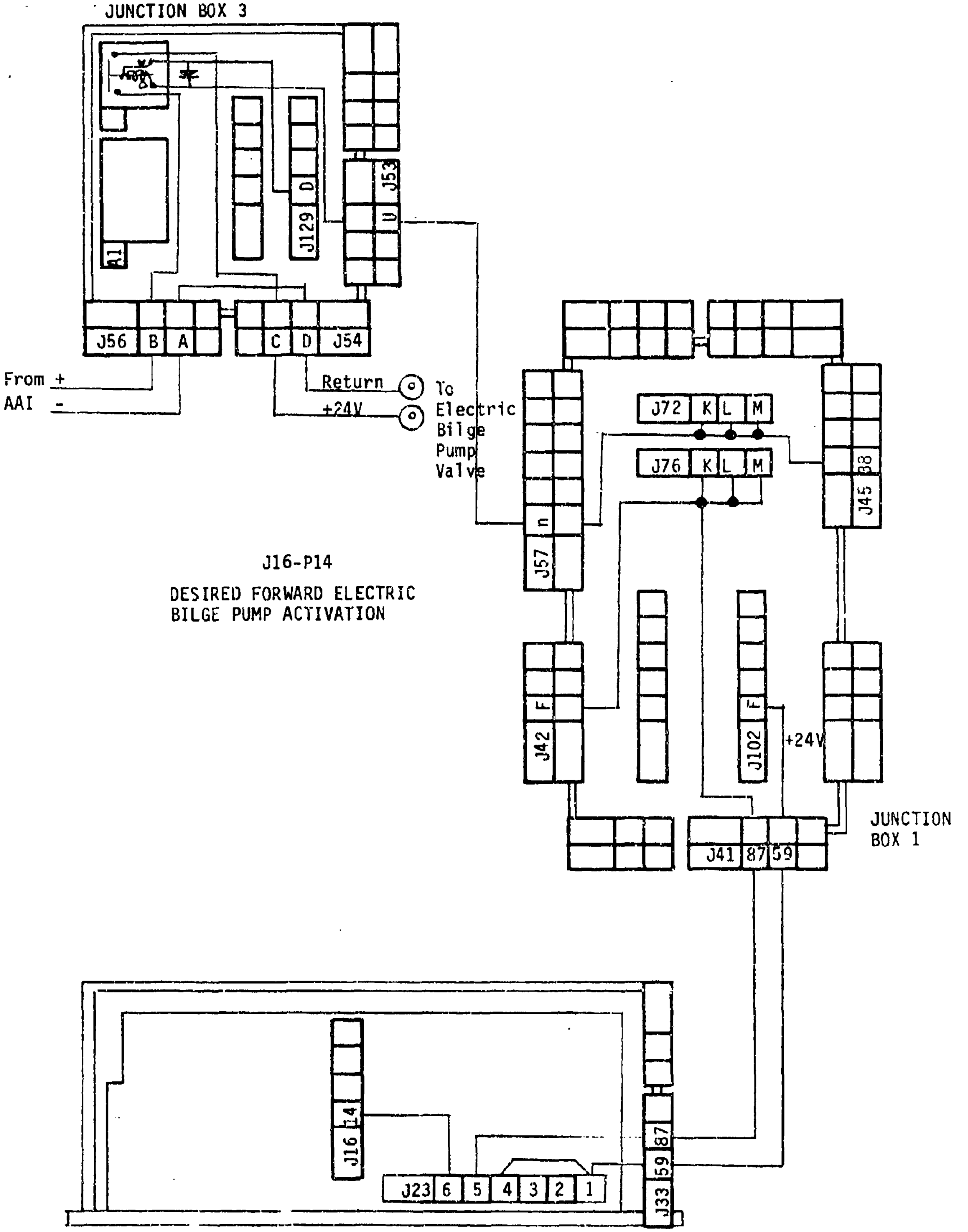



GND
J118
K
J139
A
Return
TO STARBOARD CLOCKWISE VALVE
+24V
J119
K
Z
J63
J148
X1
X2
A1
A2
JUNCTION BOX 2
J51
W
J16/P11
DESIRED STARBOARD BUCKET CLOCKWISE VALVE STATUS
J49
W
J92
G
H
J
77
J45
JUNCTION BOX 1
J41
66
11
J16
66
J33

J16-P12

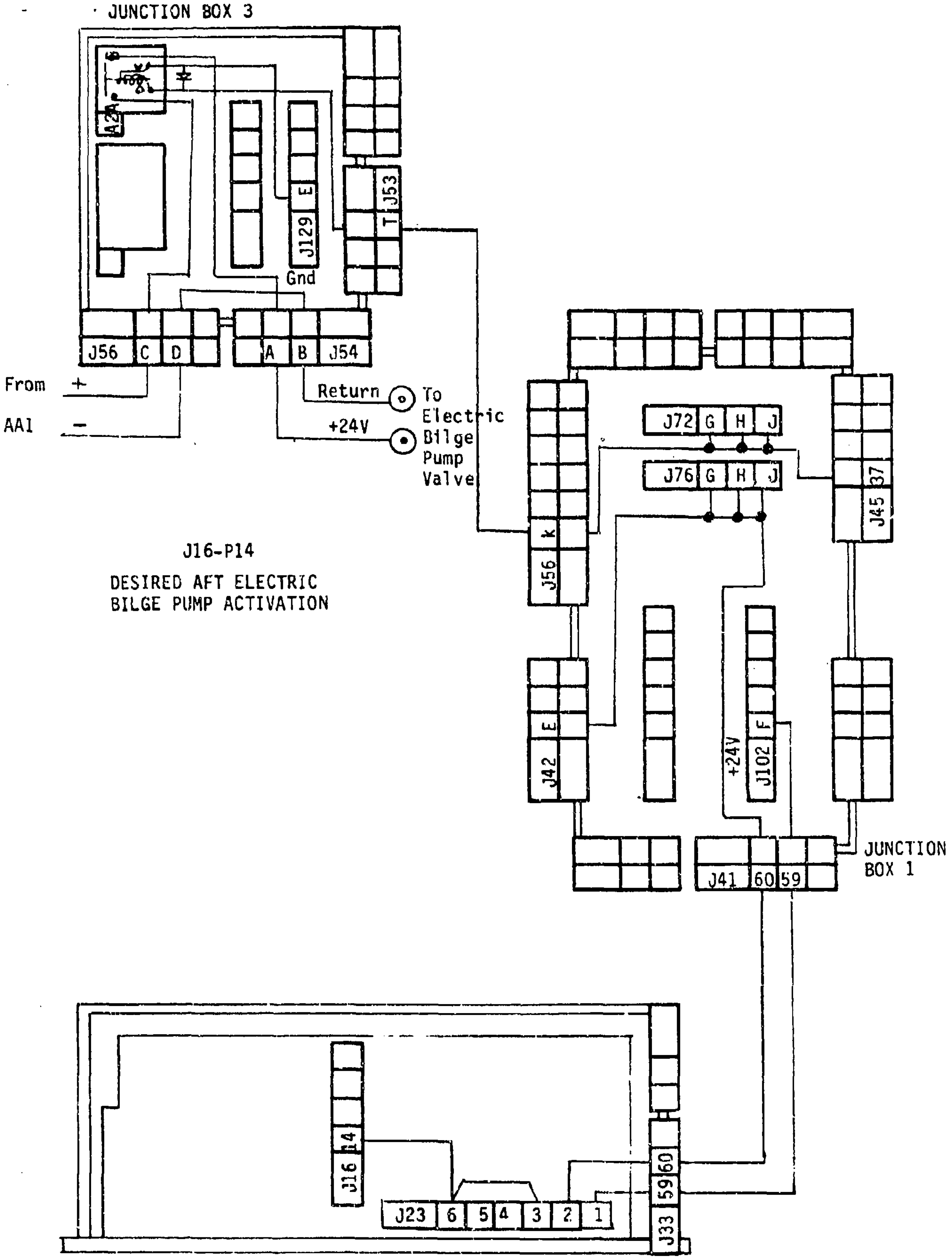
DESIRED STARBOARD BUCKET COUNTERCLOCKWISE VALVE STATUS





J16-P14

DESIRED FORWARD ELECTRIC BILGE PUMP ACTIVATION

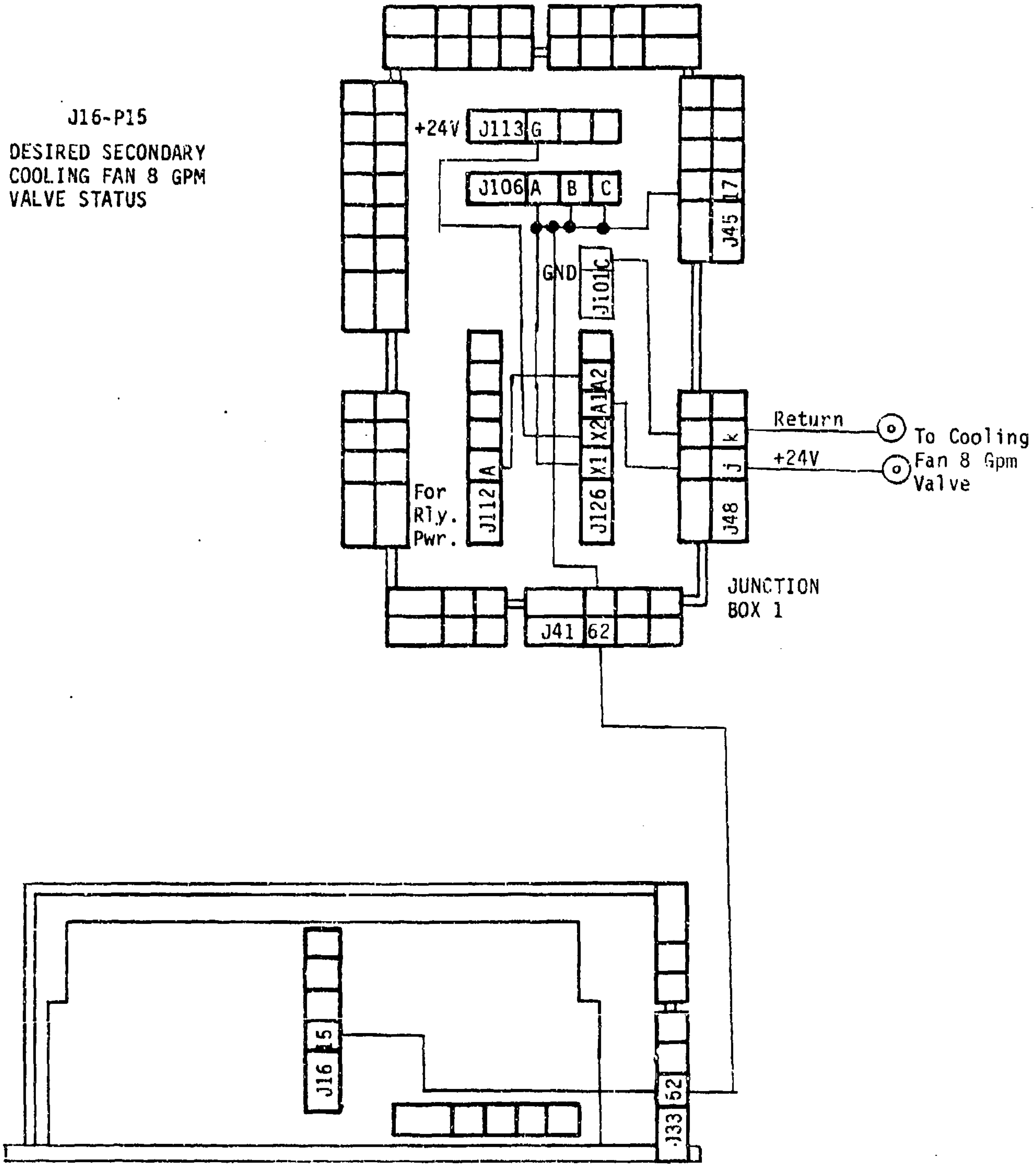


J16-P14

DESIRED AFT ELECTRIC BILGE PUMP ACTIVATION

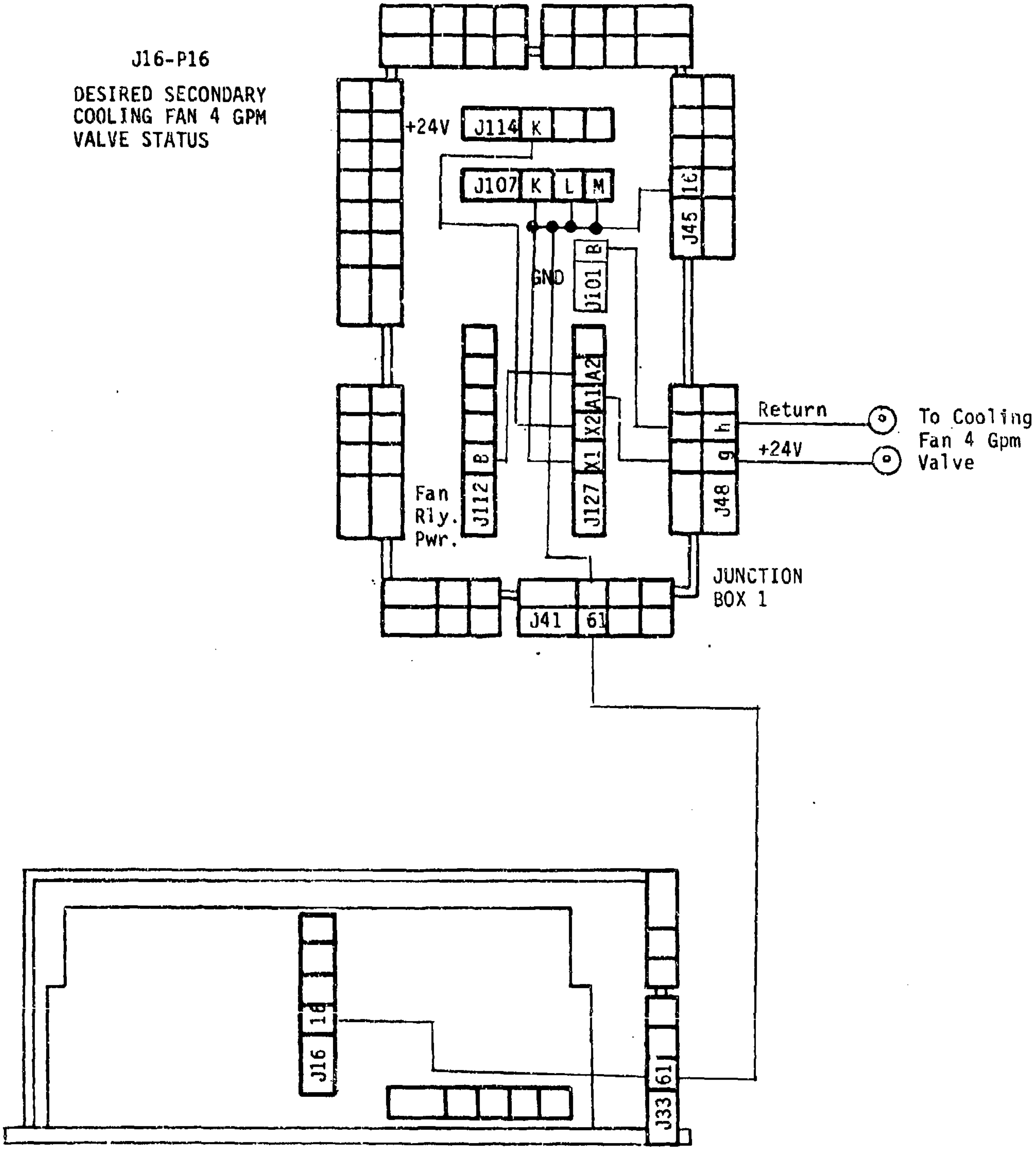
J16-P15

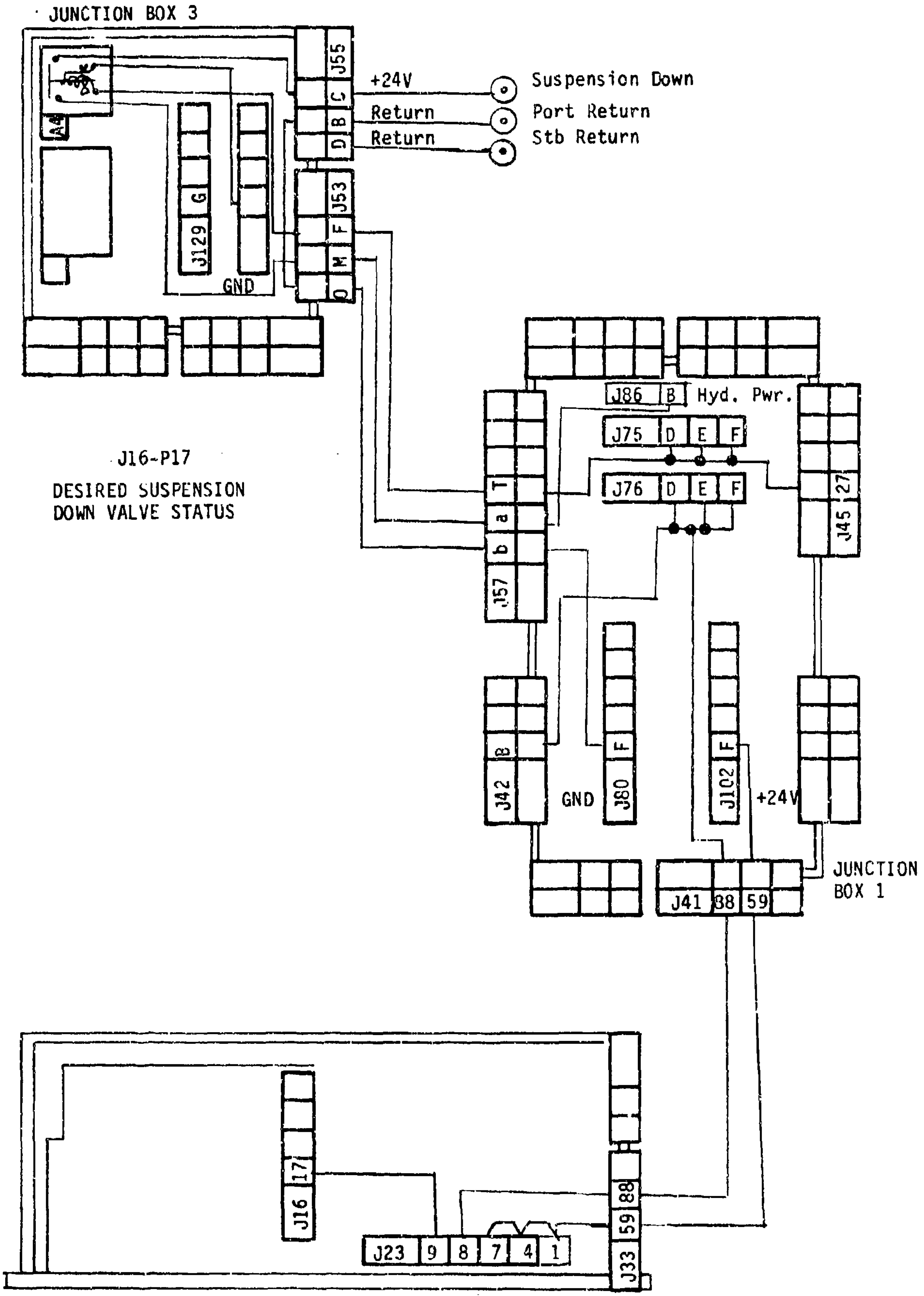
DESIRED SECONDARY COOLING FAN 8 GPM VALVE STATUS

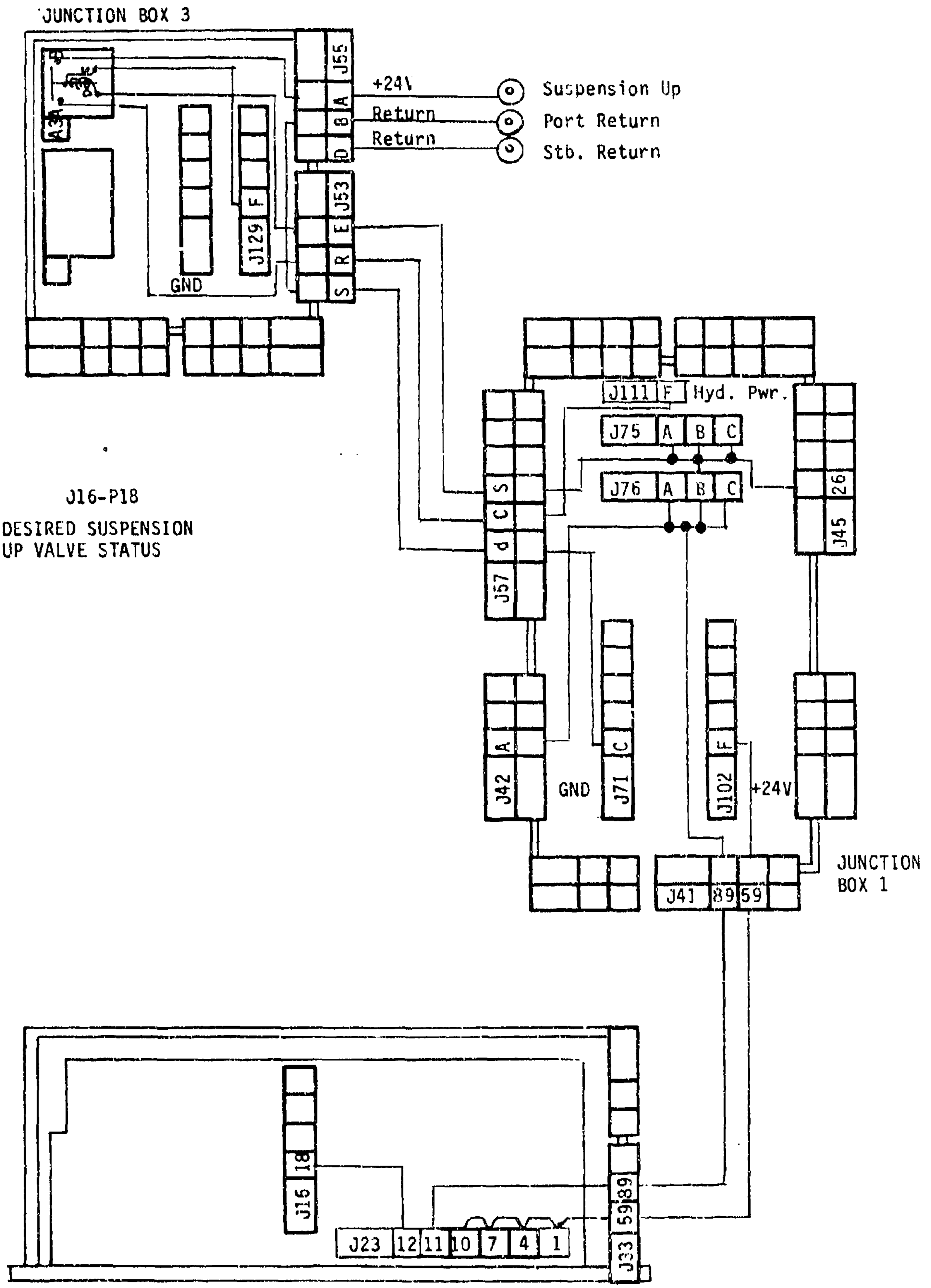


J16-P16

DESIRED SECONDARY COOLING FAN 4 GPM VALVE STATUS




JUNCTION BOX 3
J55
C
B
D
+24V
Return
Return
Suspension Down
Port Return
Stb Return
A4
G
J129
GND
J53
F
M
Q
J16-P17
DESIRED SUSPENSION DOWN VALVE STATUS
J86
B
Hyd. Pwr.
J75
D
E
F
J76
D
E
F
T
a
b
J57
27
J45
B
J42
GND
F
J80
F
J102
+24V
JUNCTION BOX 1
J41
88
59
17
J16
J23
9
8
7
4
1
88
59
J33

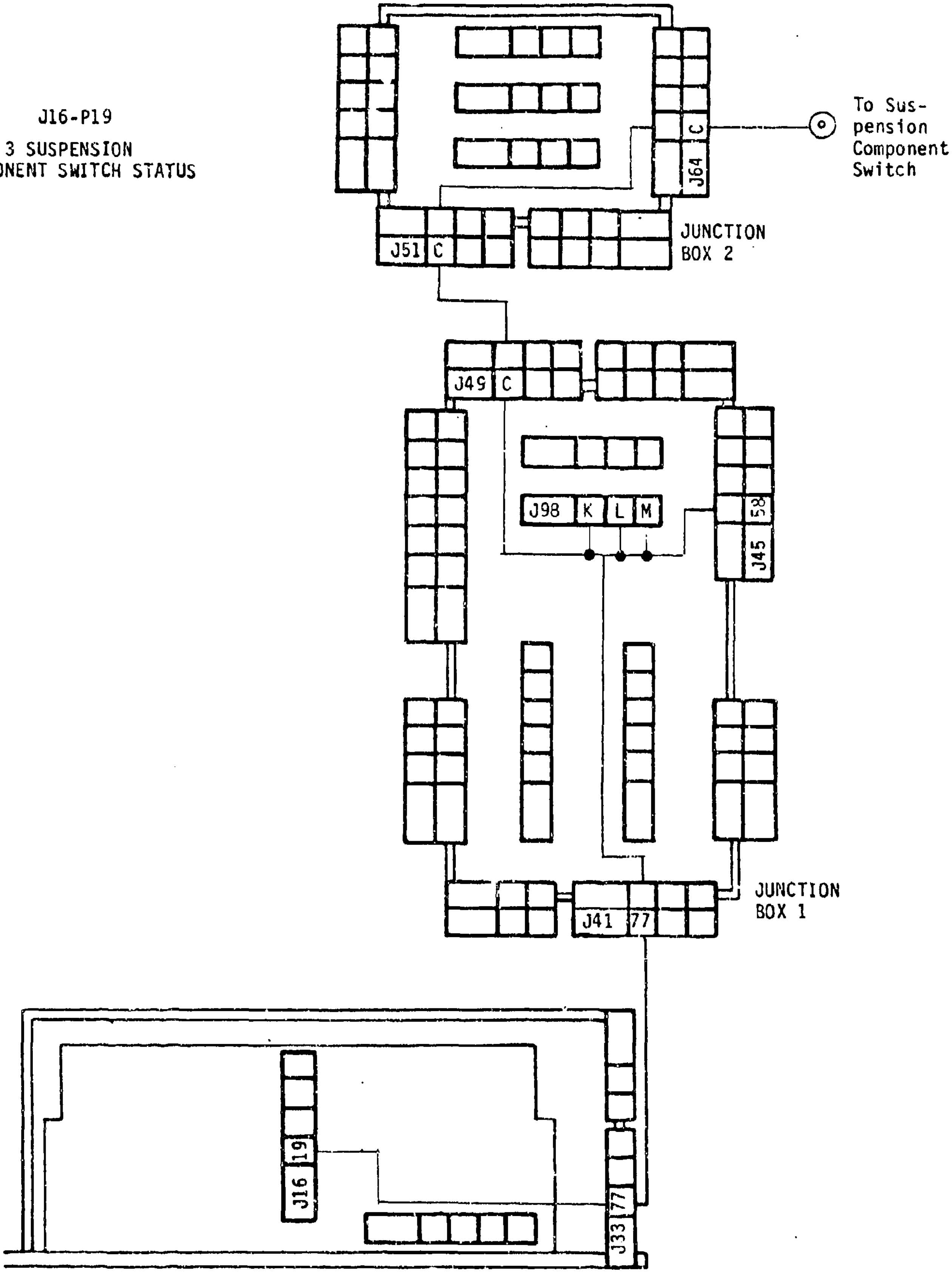


J16-P18

DESIRED SUSPENSION
UP VALVE STATUS

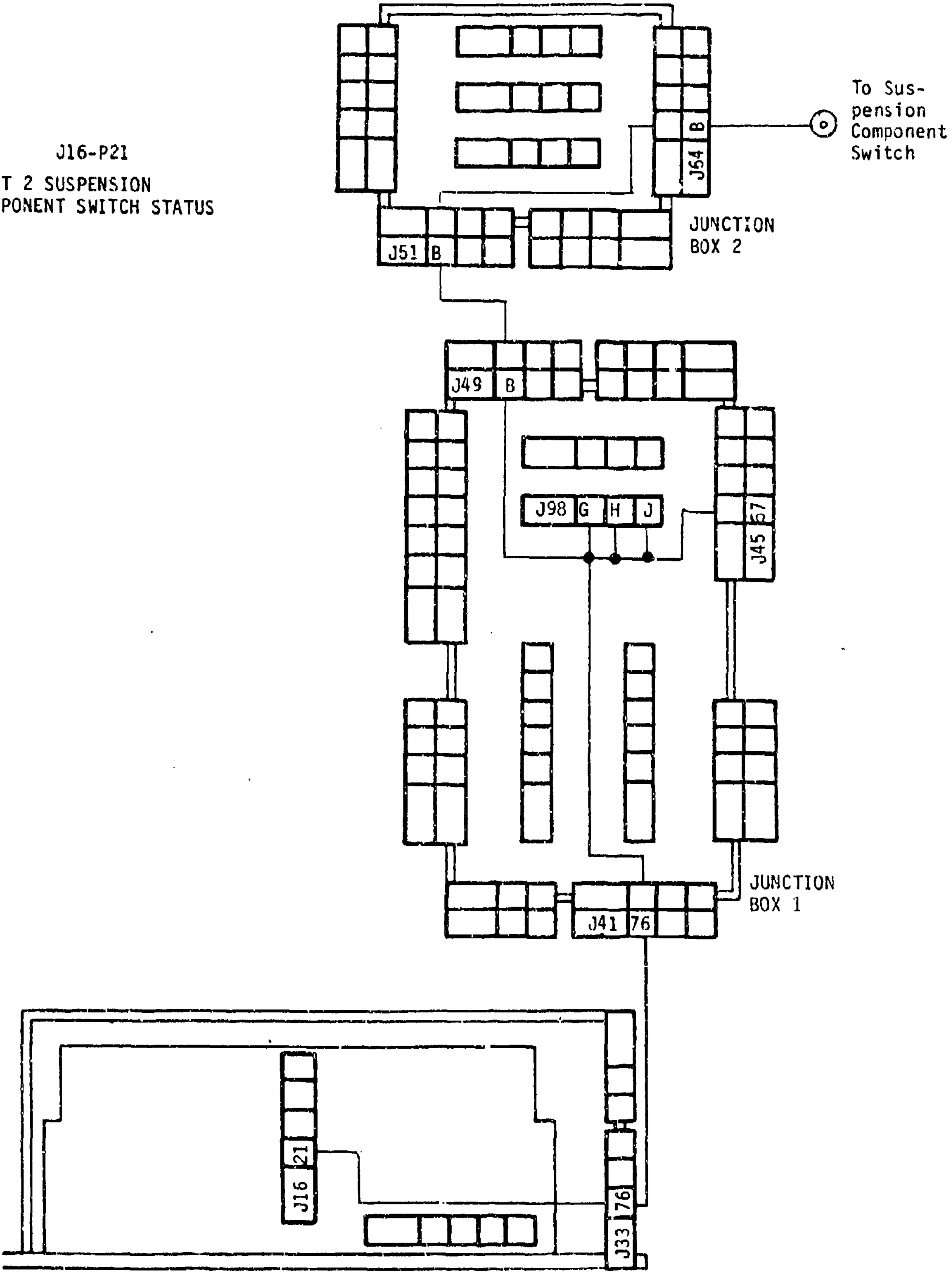
J16-P19

PORT 3 SUSPENSION
COMPONENT SWITCH STATUS



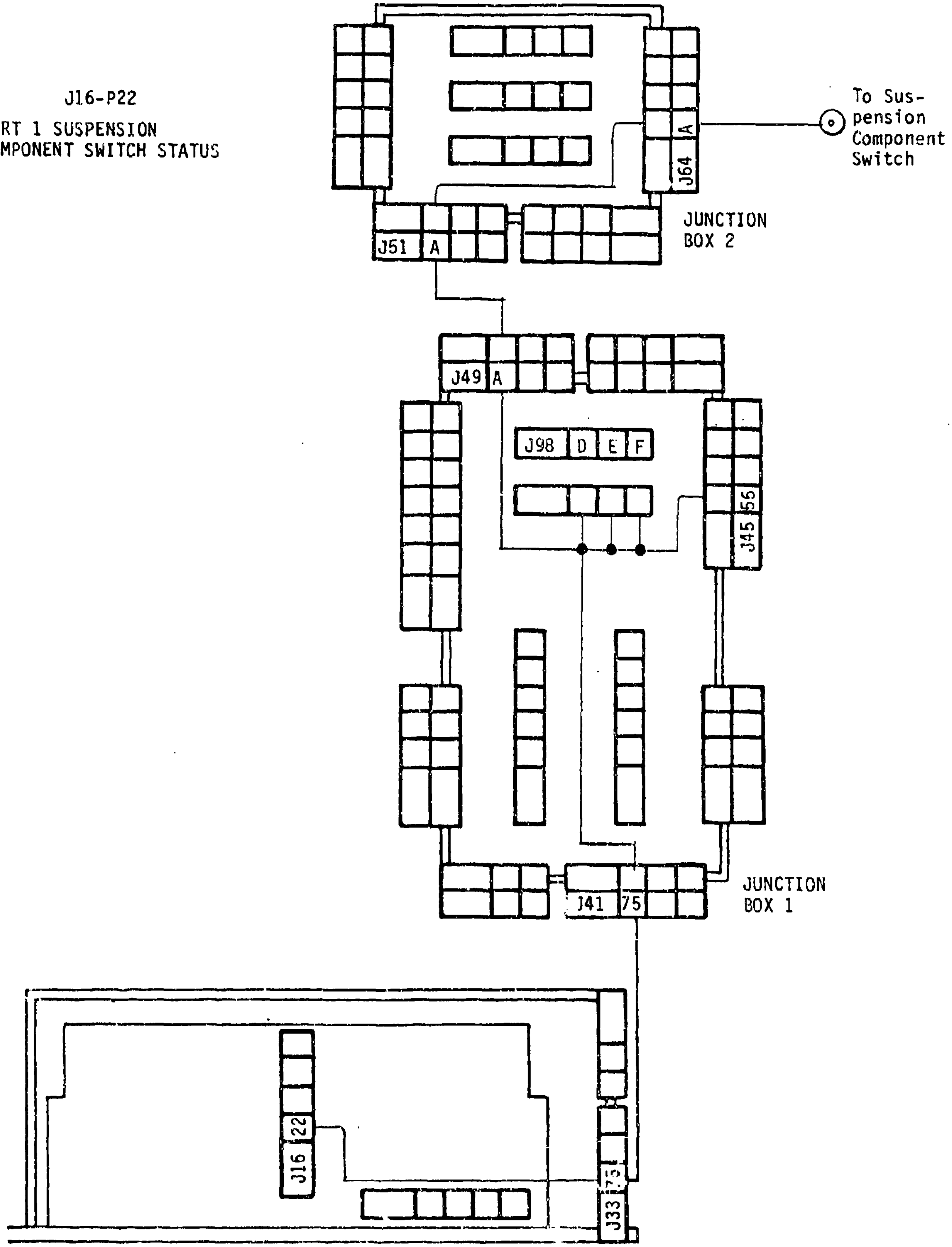
J16-P21

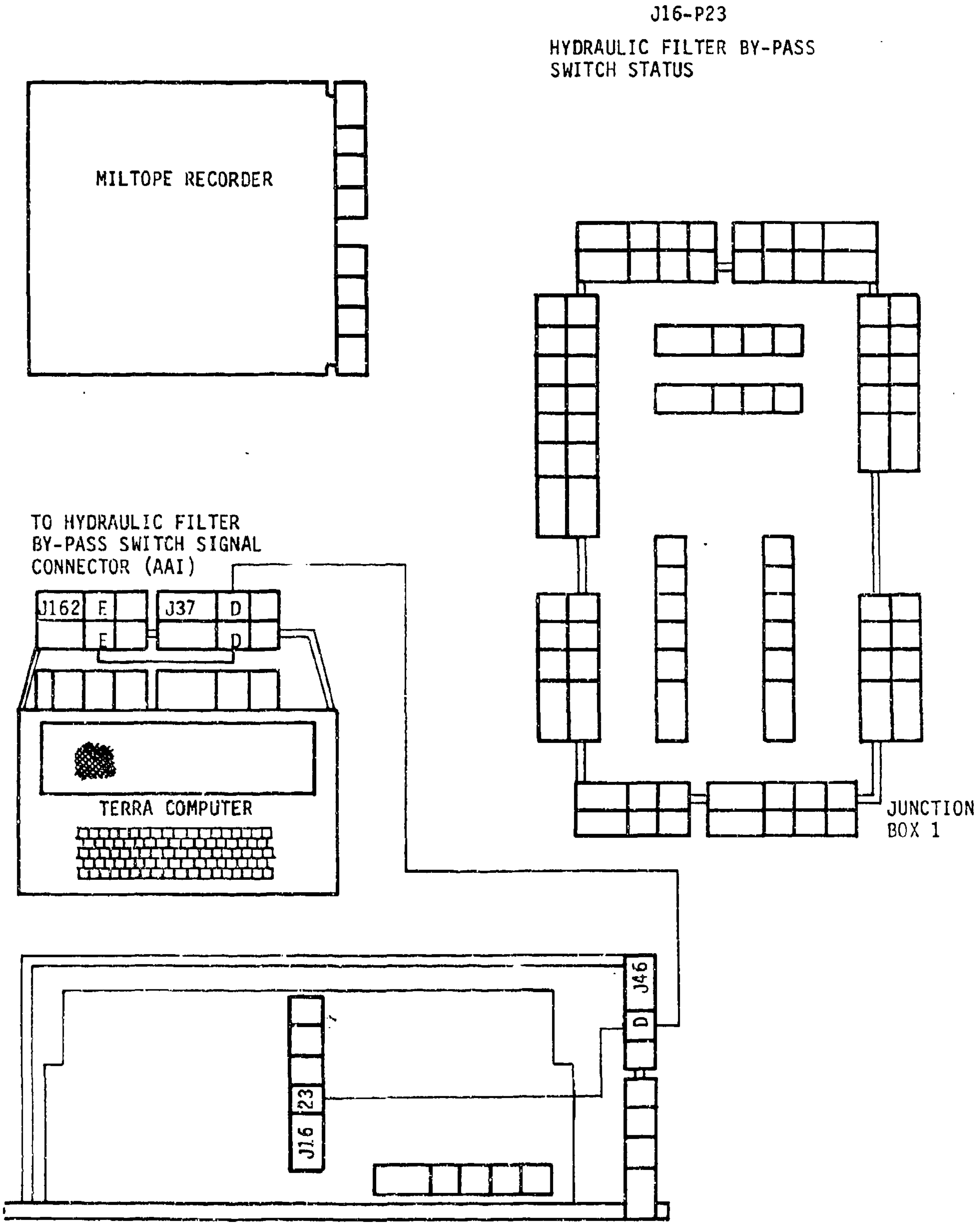
PORT 2 SUSPENSION COMPONENT SWITCH STATUS



J16-P22

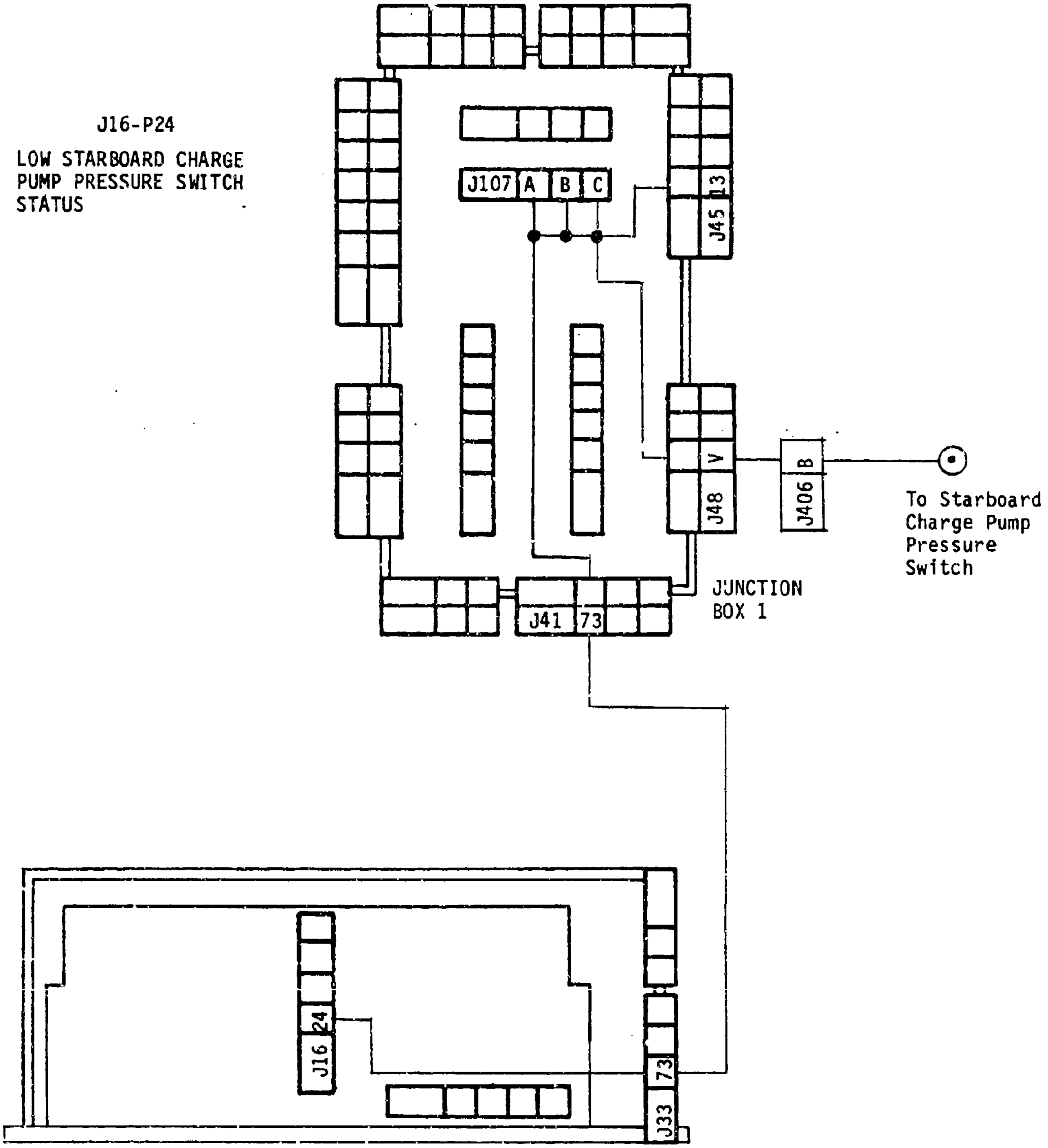
PORT 1 SUSPENSION COMPONENT SWITCH STATUS




J16-P23
HYDRAULIC FILTER BY-PASS
SWITCH STATUS
MILTOPE RECORDER
TO HYDRAULIC FILTER
BY-PASS SWITCH SIGNAL
CONNECTOR (AAI)
J162
E
E
J37
D
D
TERRA COMPUTER
JUNCTION
BOX 1
J46
D
J16
23

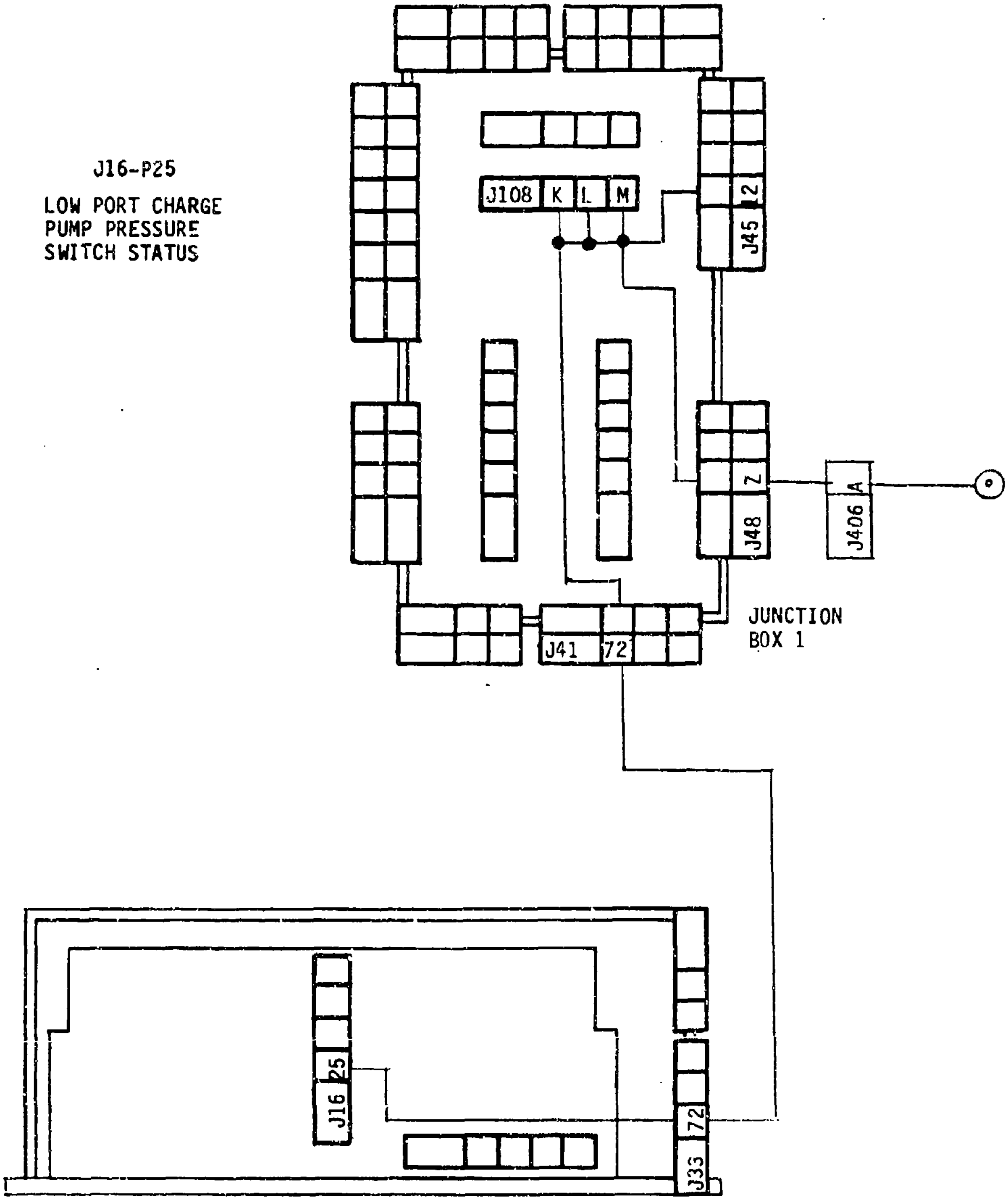
J16-P24

LOW STARBOARD CHARGE PUMP PRESSURE SWITCH STATUS



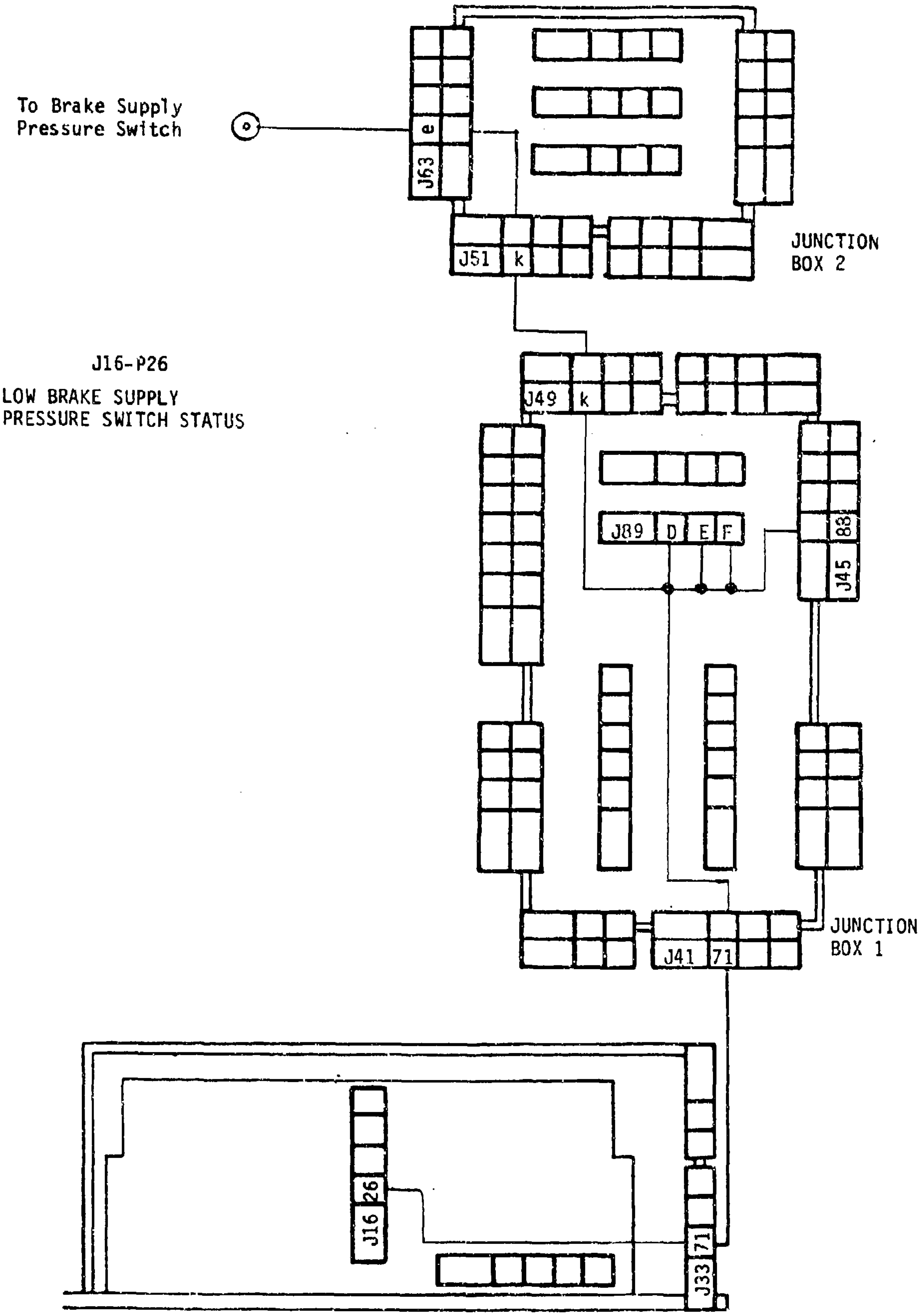
J16-P25

LOW PORT CHARGE PUMP PRESSURE SWITCH STATUS



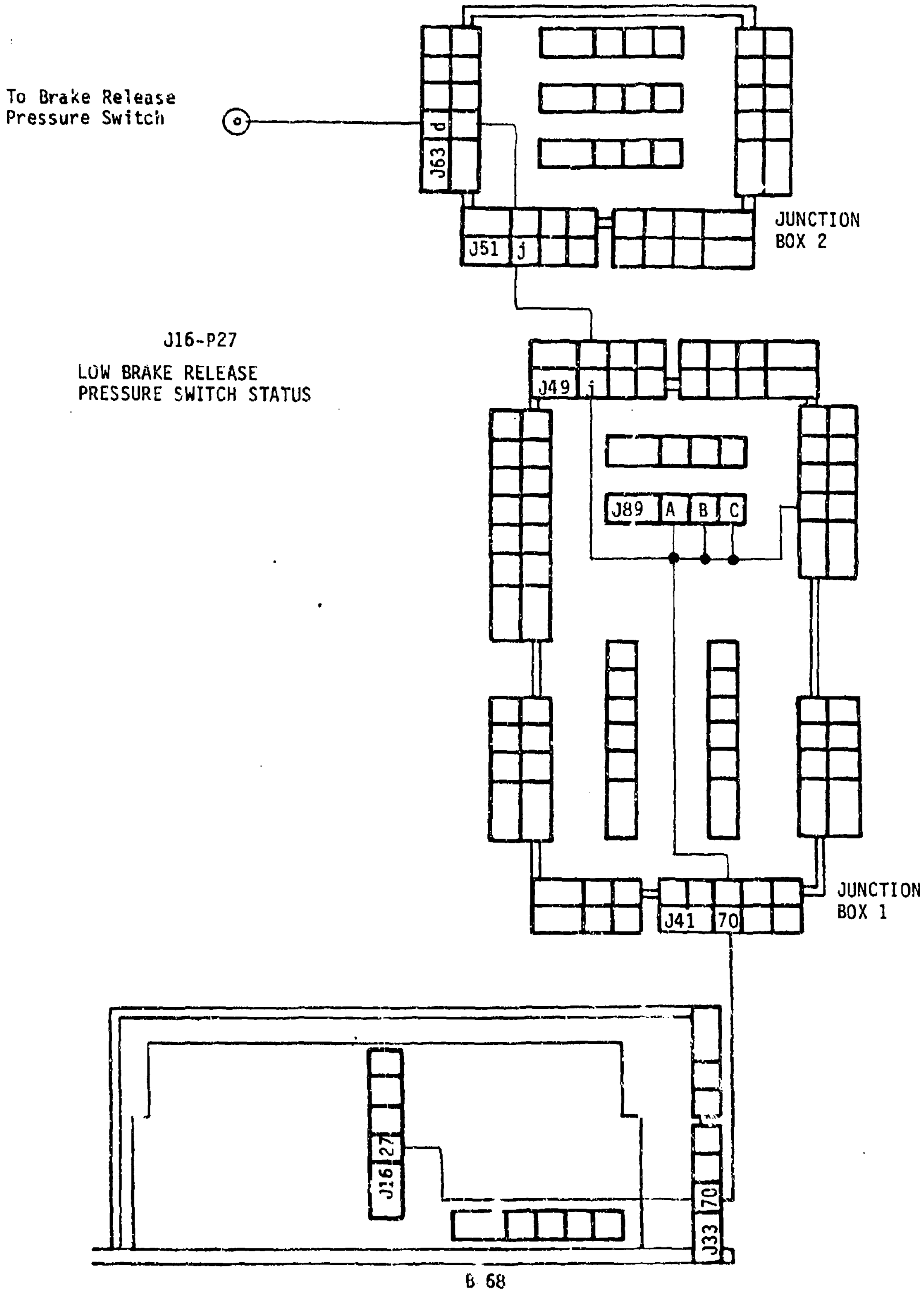
J16-P26

LOW BRAKE SUPPLY PRESSURE SWITCH STATUS



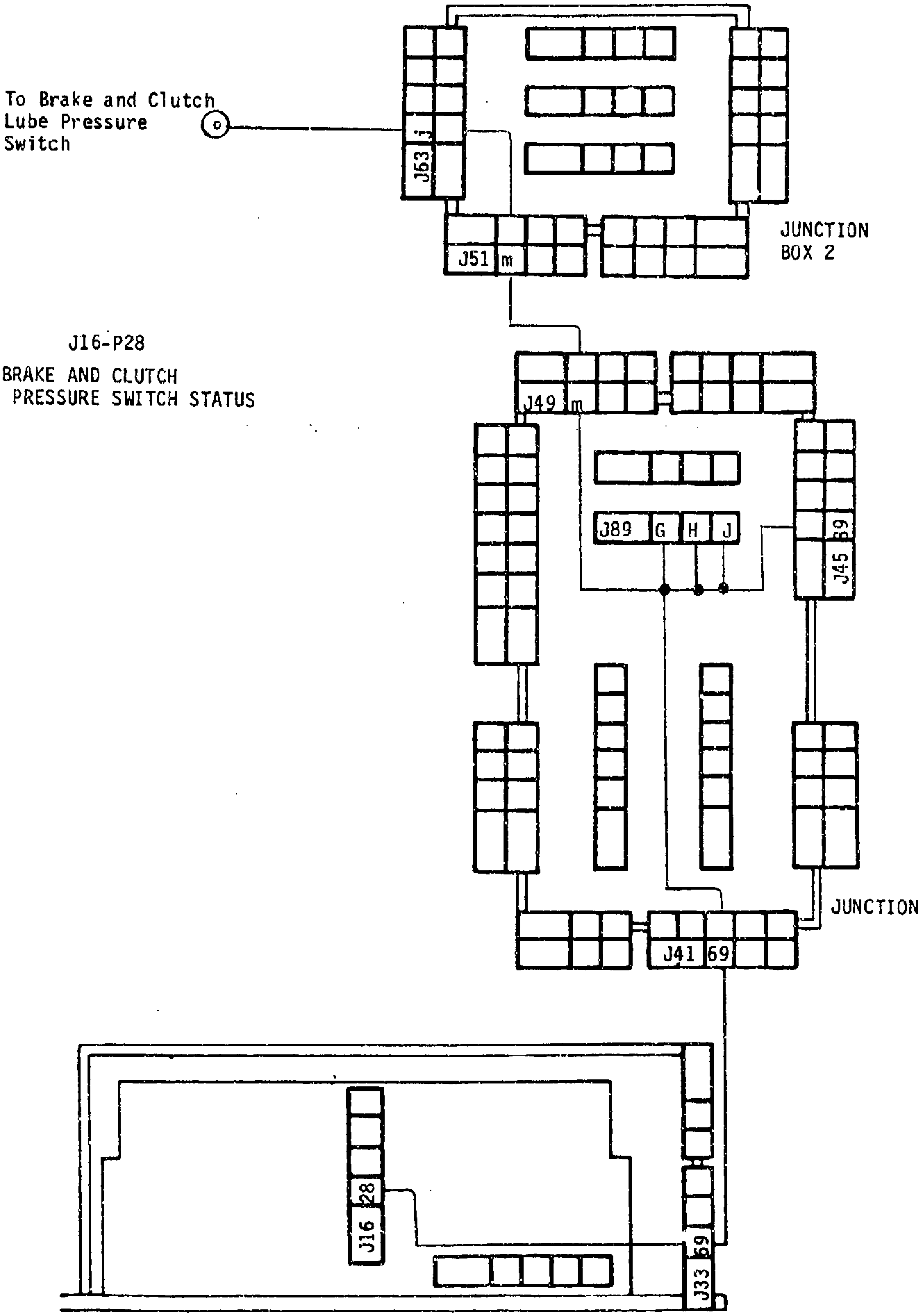
J16-P27

LOW BRAKE RELEASE PRESSURE SWITCH STATUS



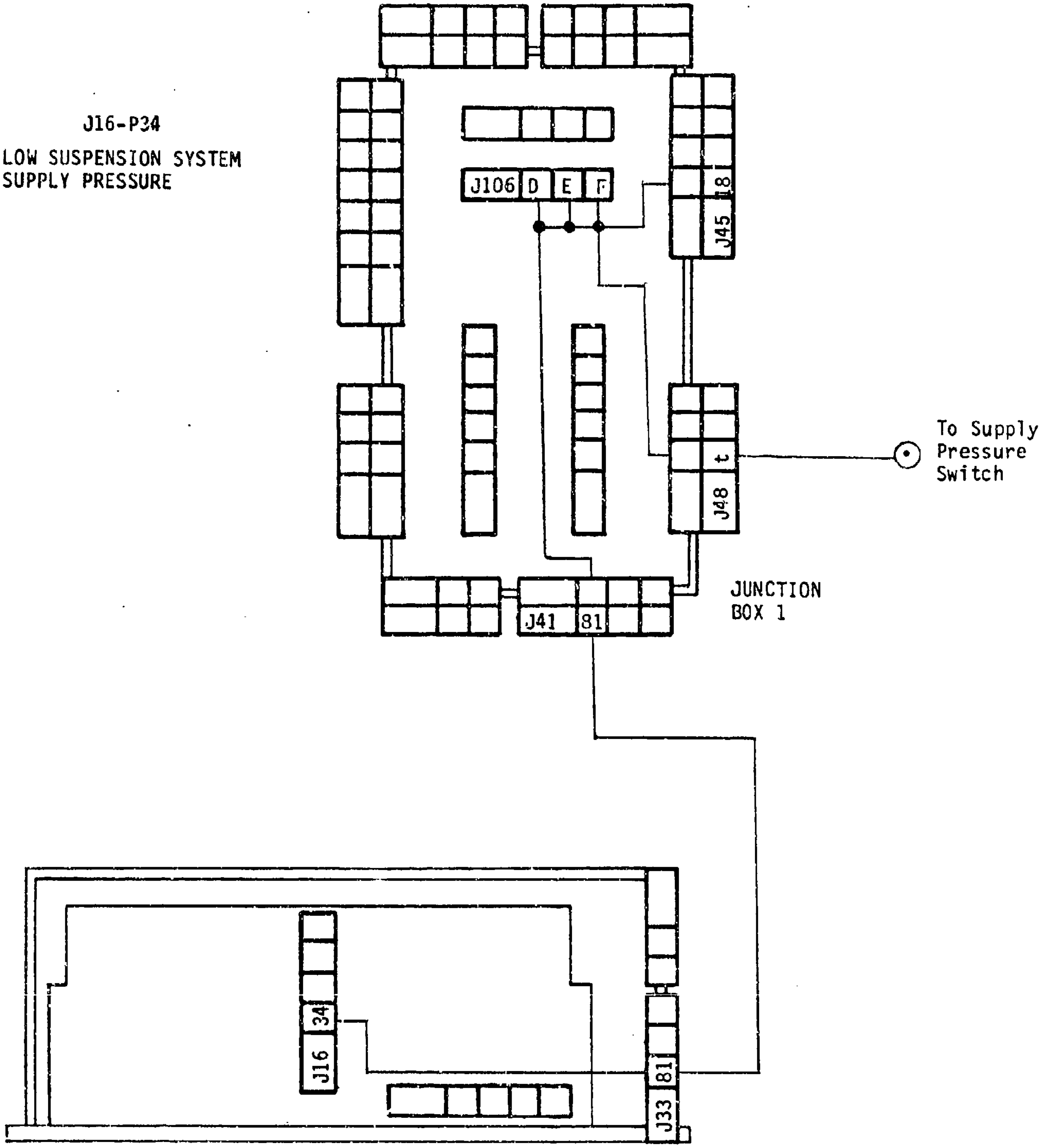
J16-P28

LOW BRAKE AND CLUTCH
LUBE PRESSURE SWITCH STATUS



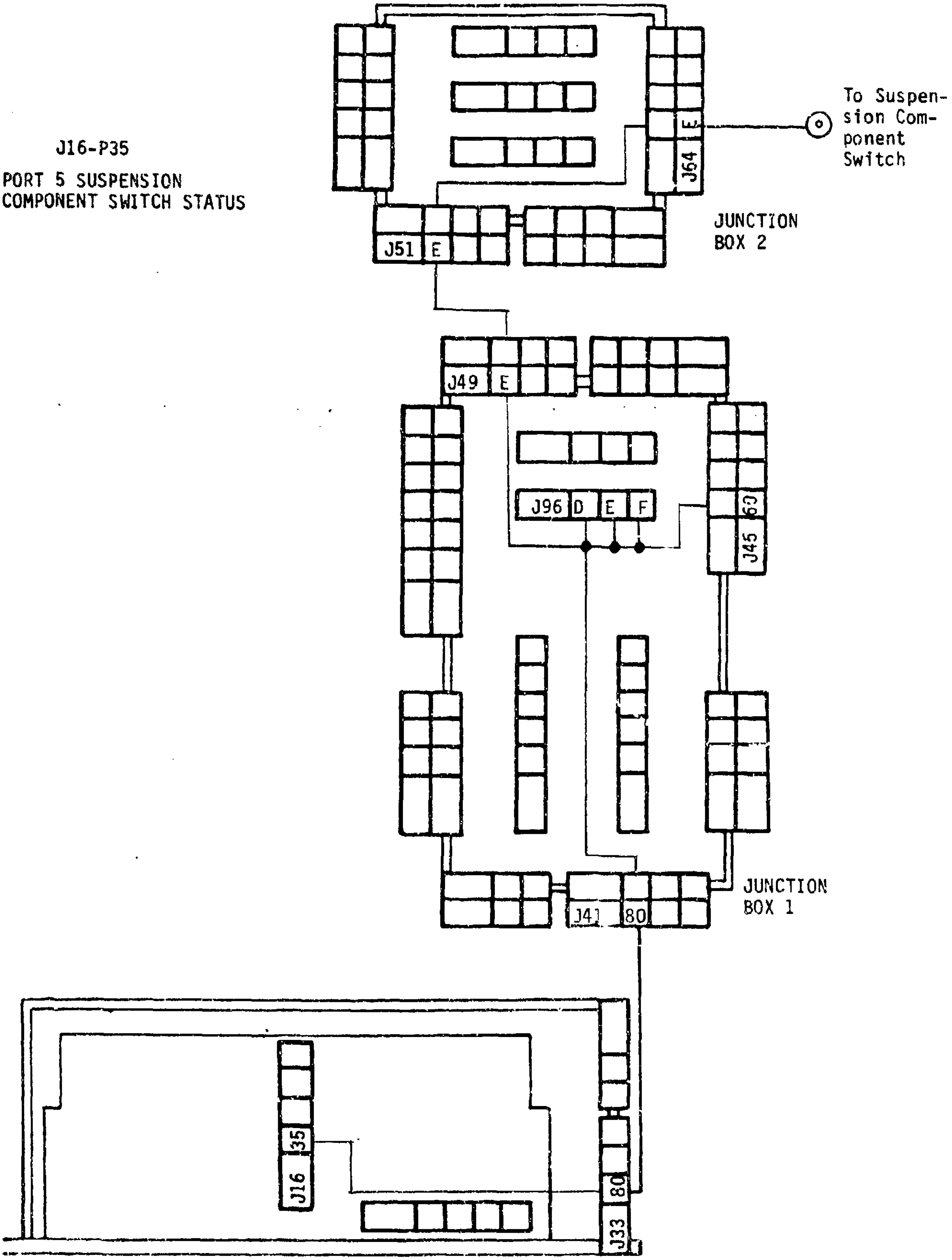
J16-P34

LOW SUSPENSION SYSTEM SUPPLY PRESSURE



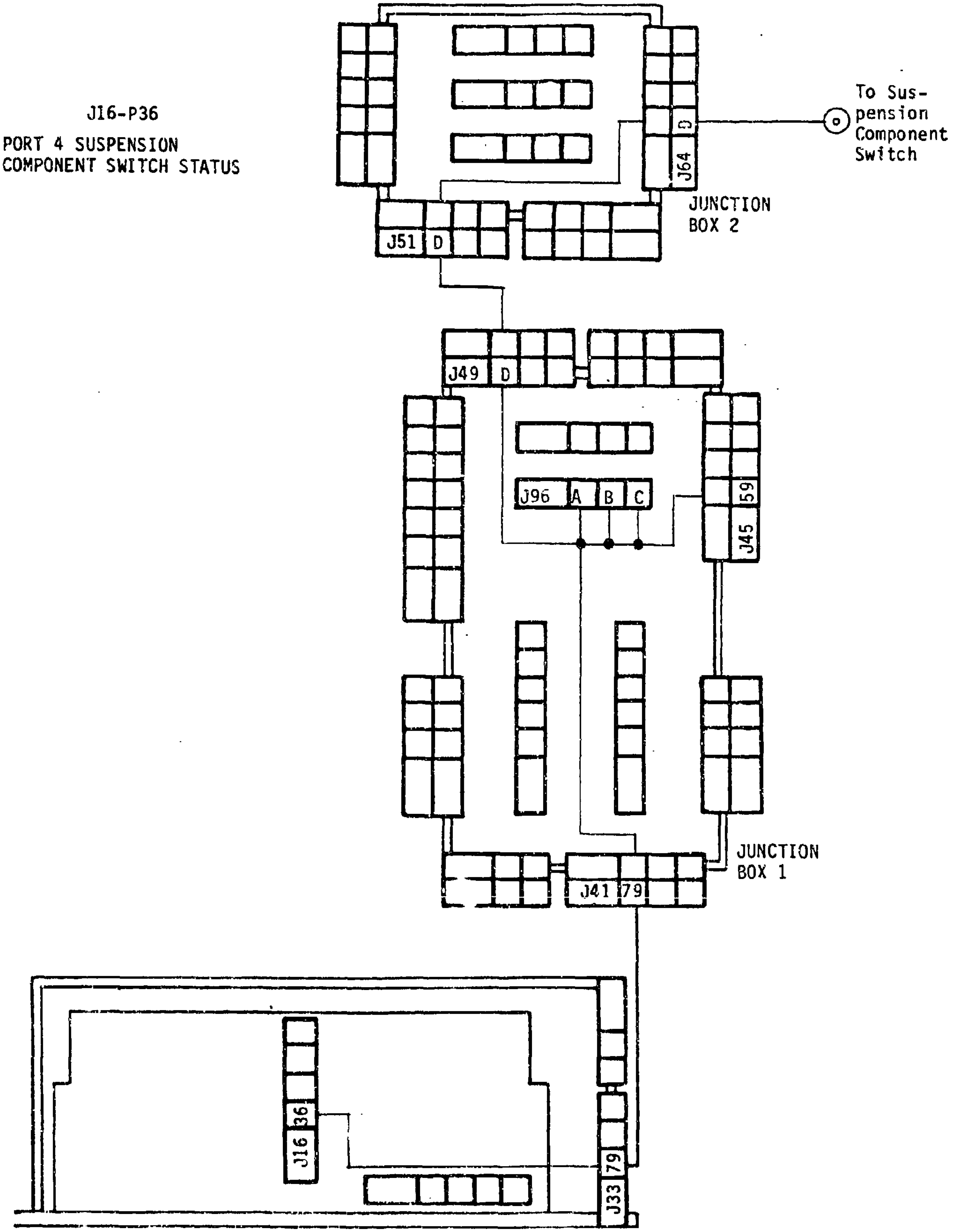
J16-P35

PORT 5 SUSPENSION COMPONENT SWITCH STATUS



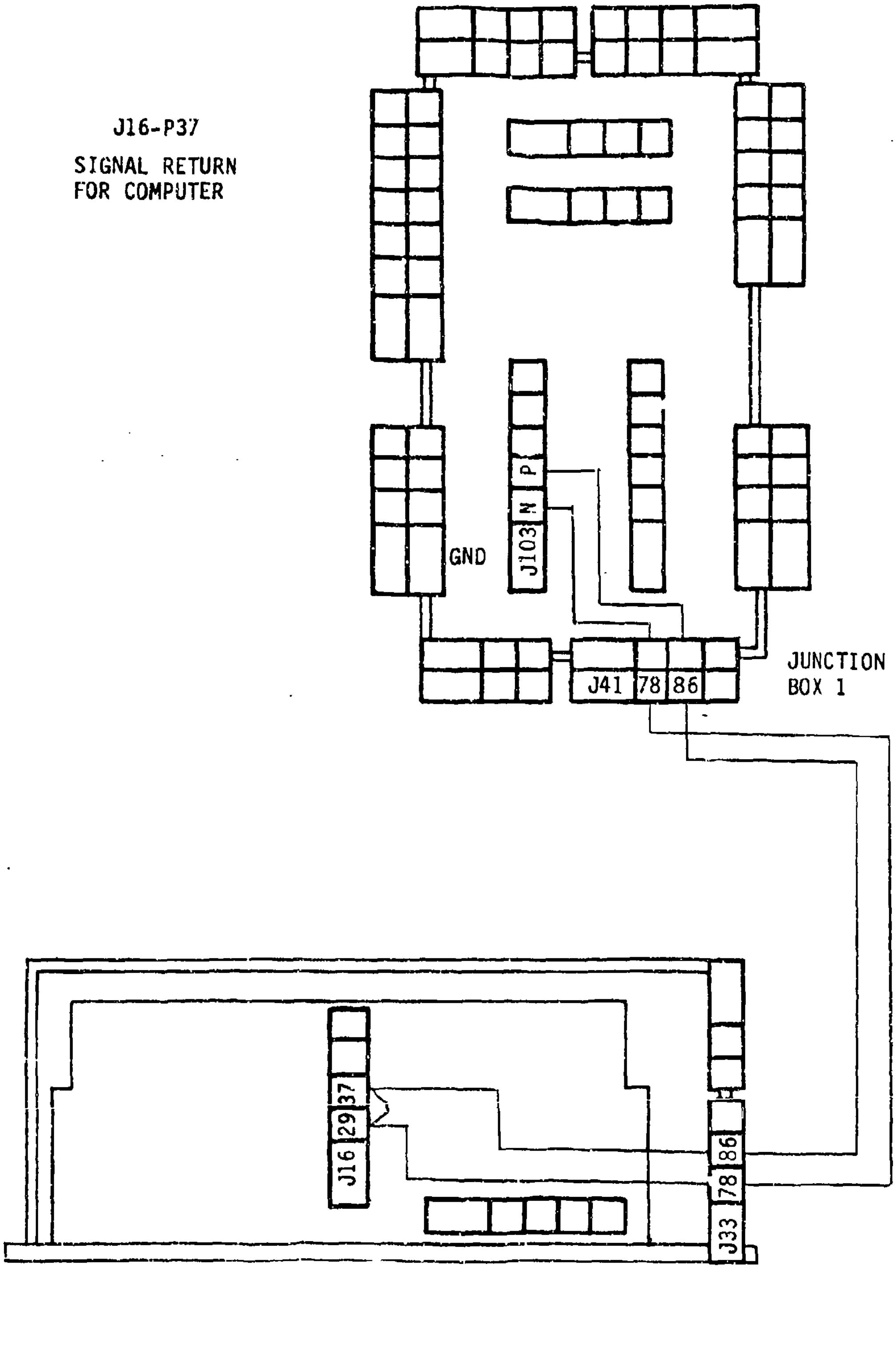
J16-P36

PORT 4 SUSPENSION COMPONENT SWITCH STATUS



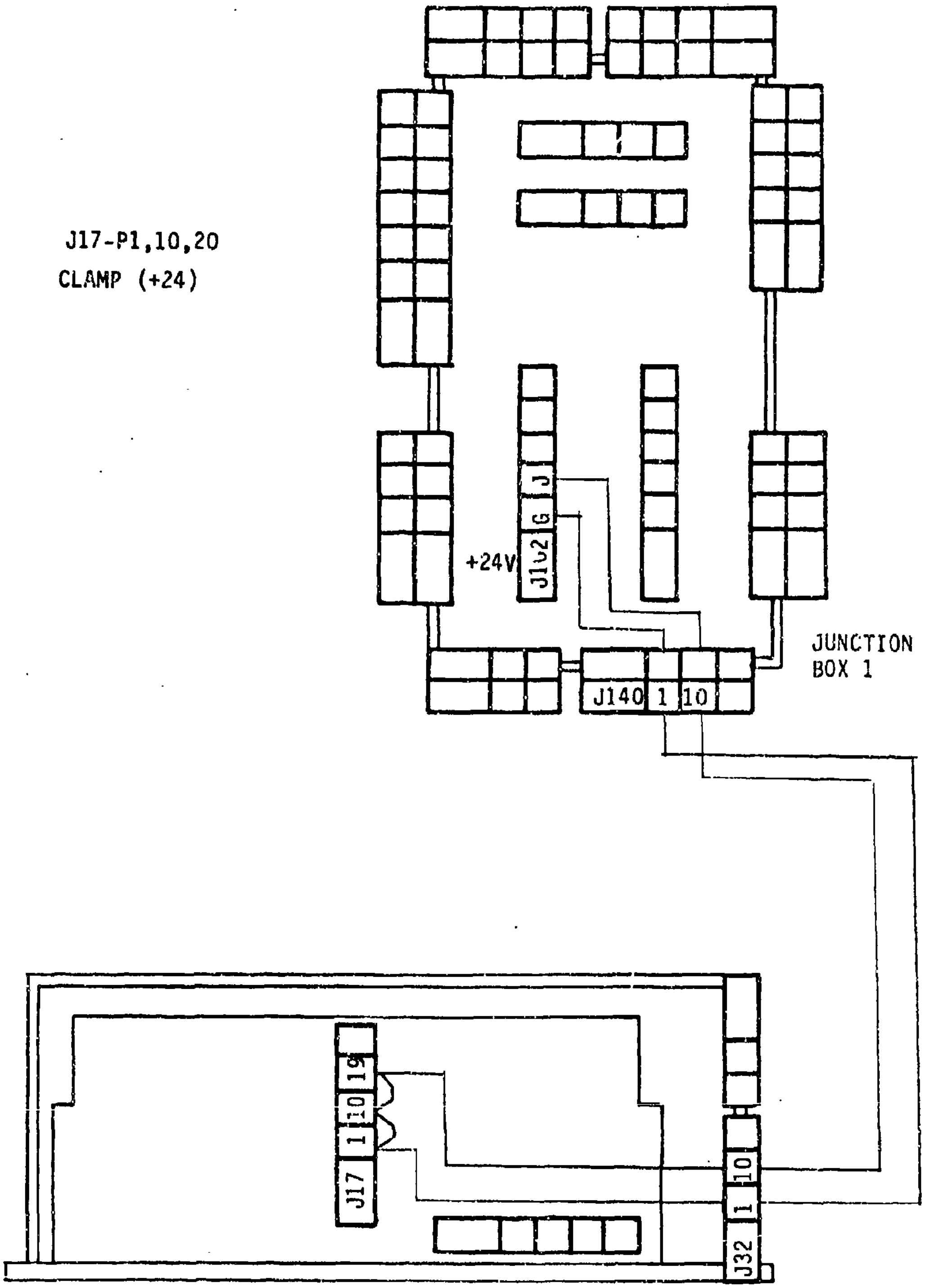
J16-P37

SIGNAL RETURN
FOR COMPUTER



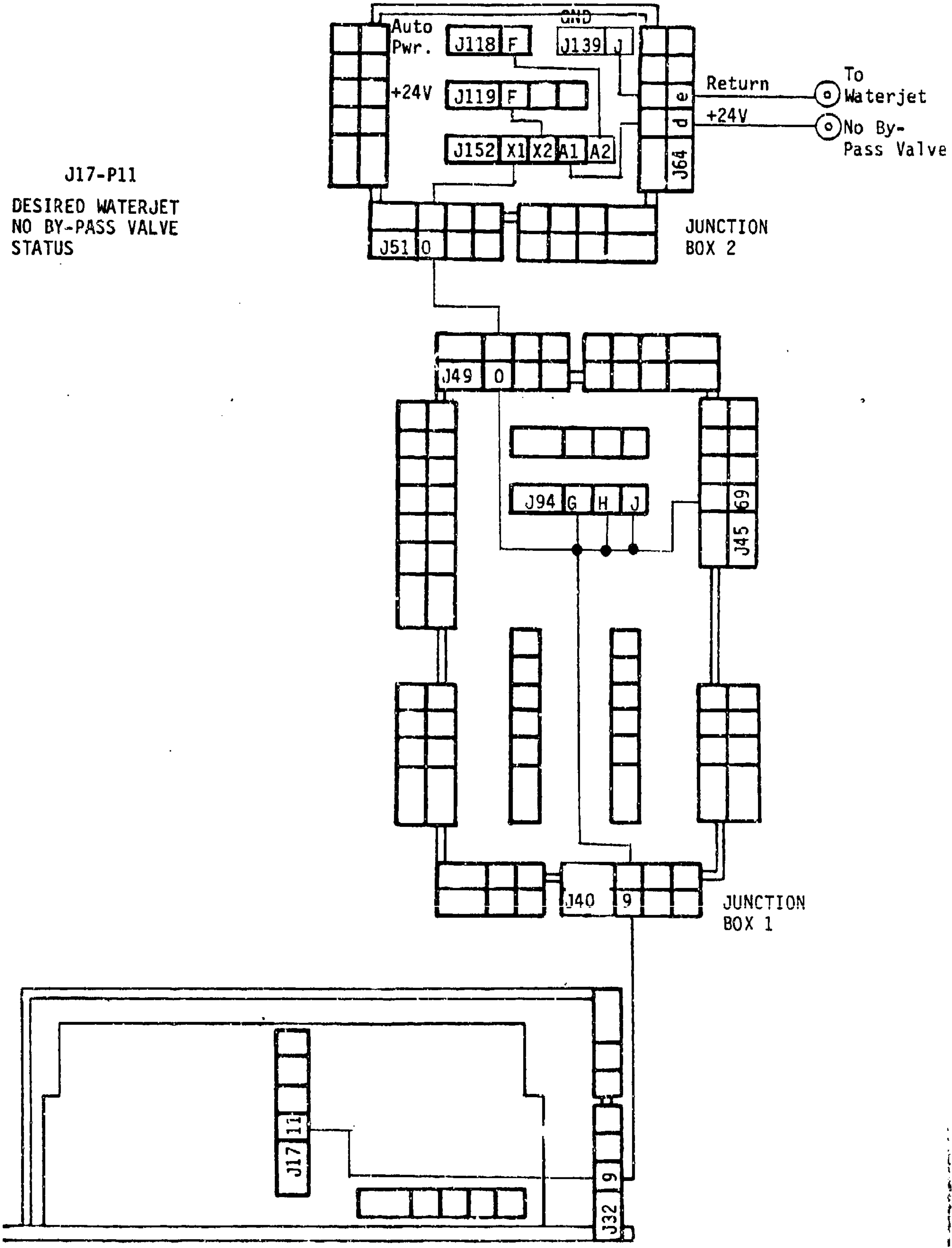
J17-P1,10,20

CLAMP (+24)



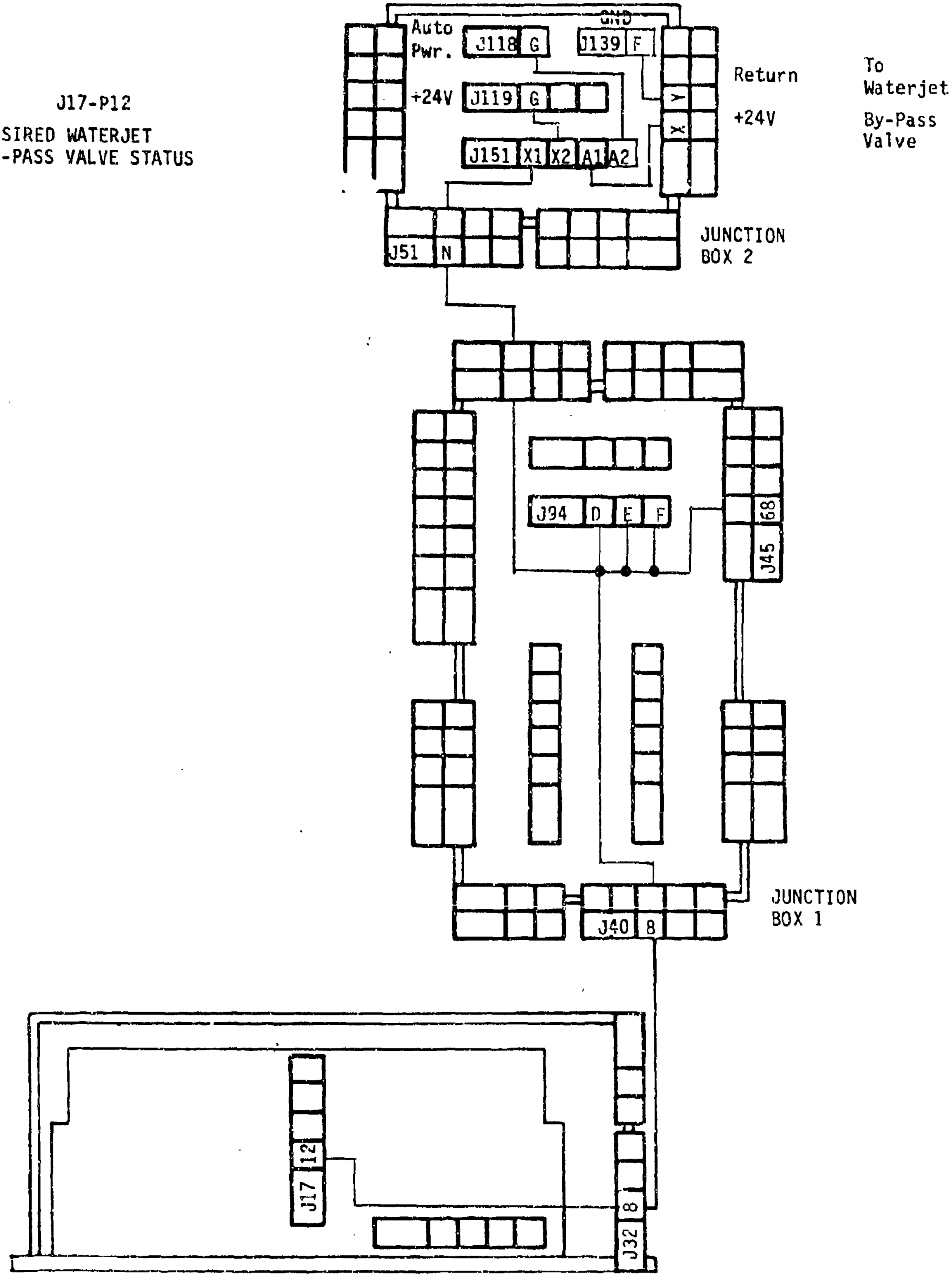
J17-P11

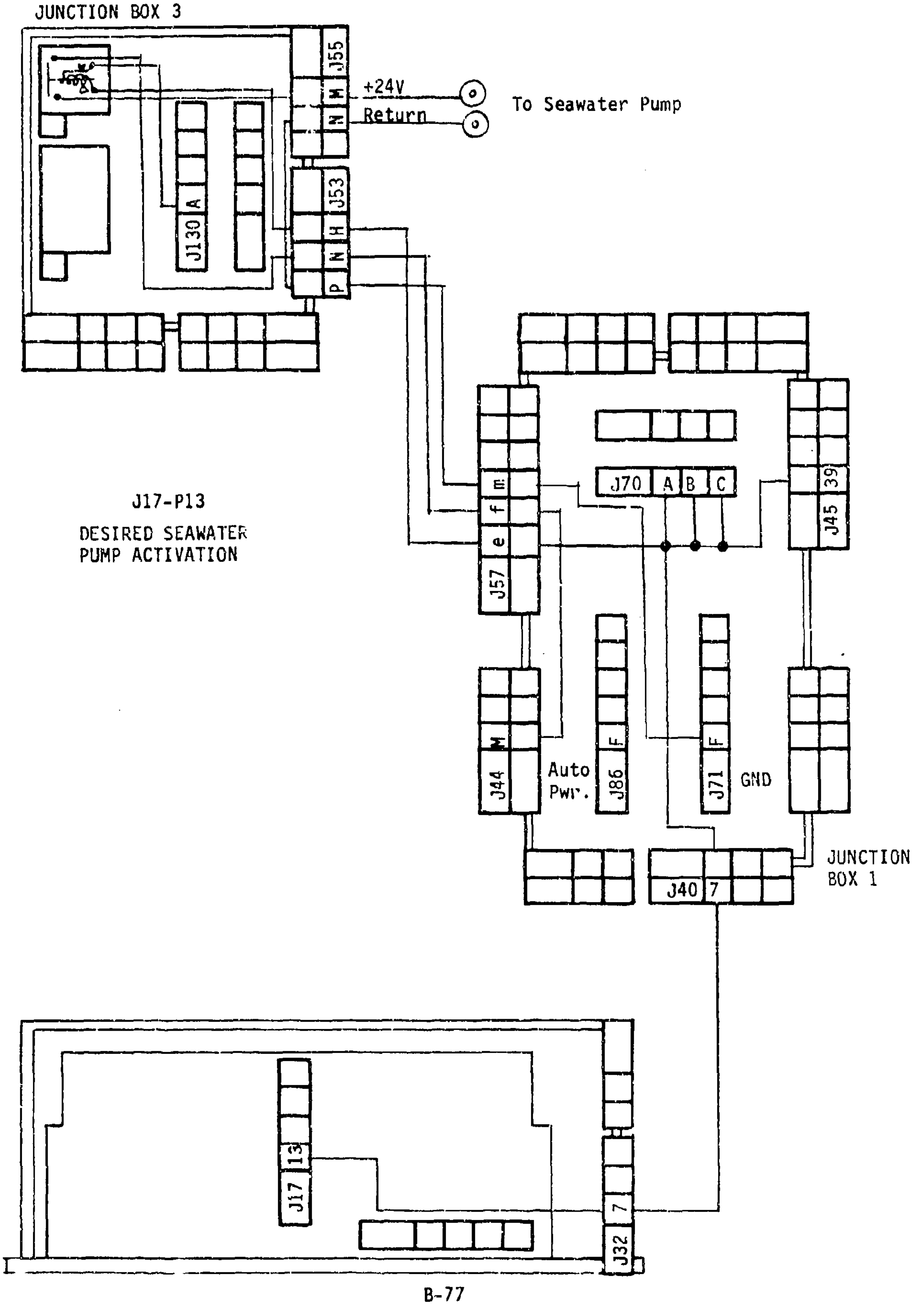
DESIRED WATERJET
NO BY-PASS VALVE
STATUS



J17-P12

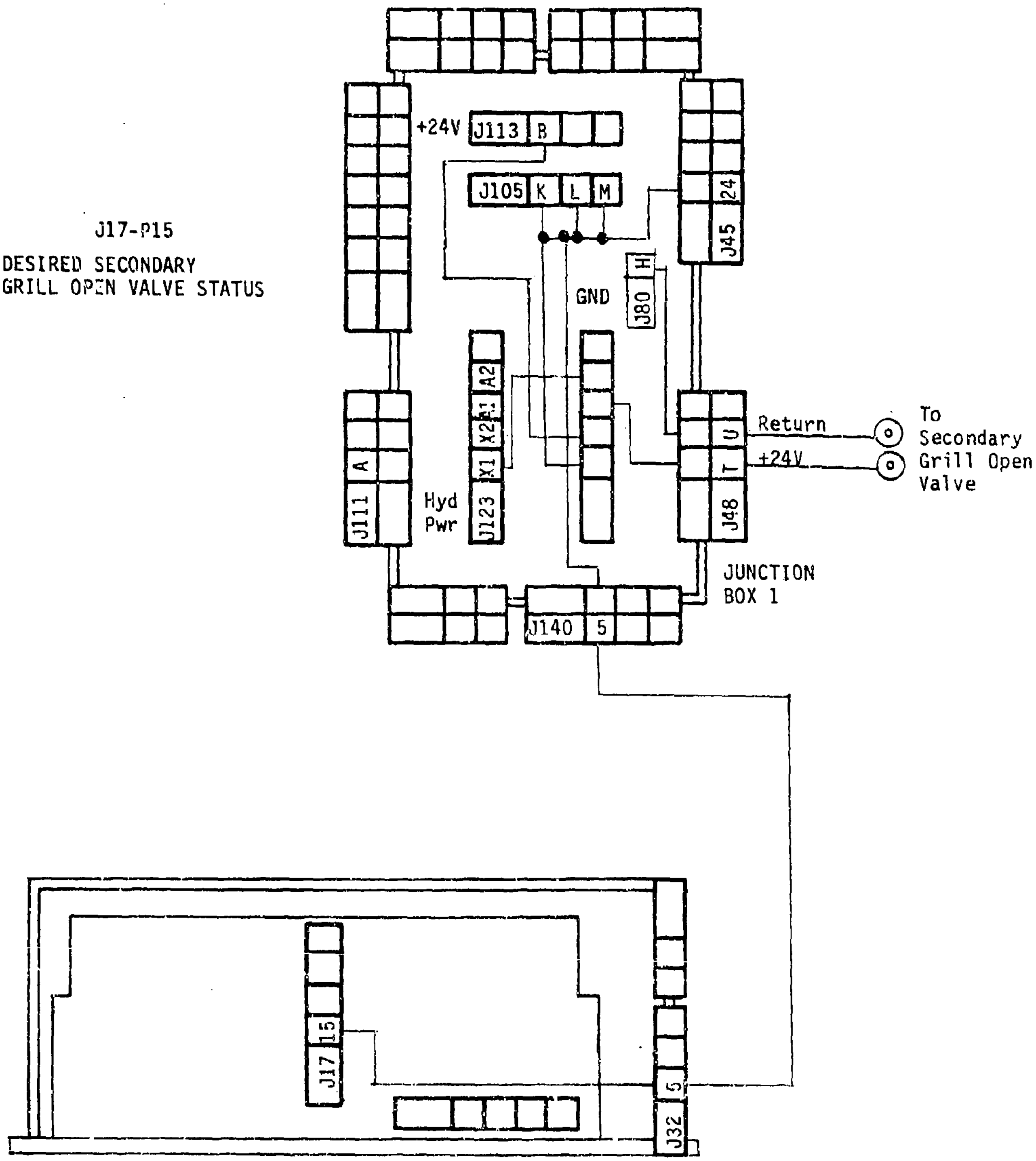
DESIRED WATERJET
BY-PASS VALVE STATUS




JUNCTION BOX 3
J55
M
N
+24V
Return
To Seawater Pump
J53
H
N
P
J130
A
J17-P13
DESIRED SEAWATER PUMP ACTIVATION
J57
m
f
e
J70
A
B
C
39
J45
J44
M
Auto Pwr.
J86
F
J71
F
GND
J40
7
JUNCTION BOX 1
J17
13
7
J32

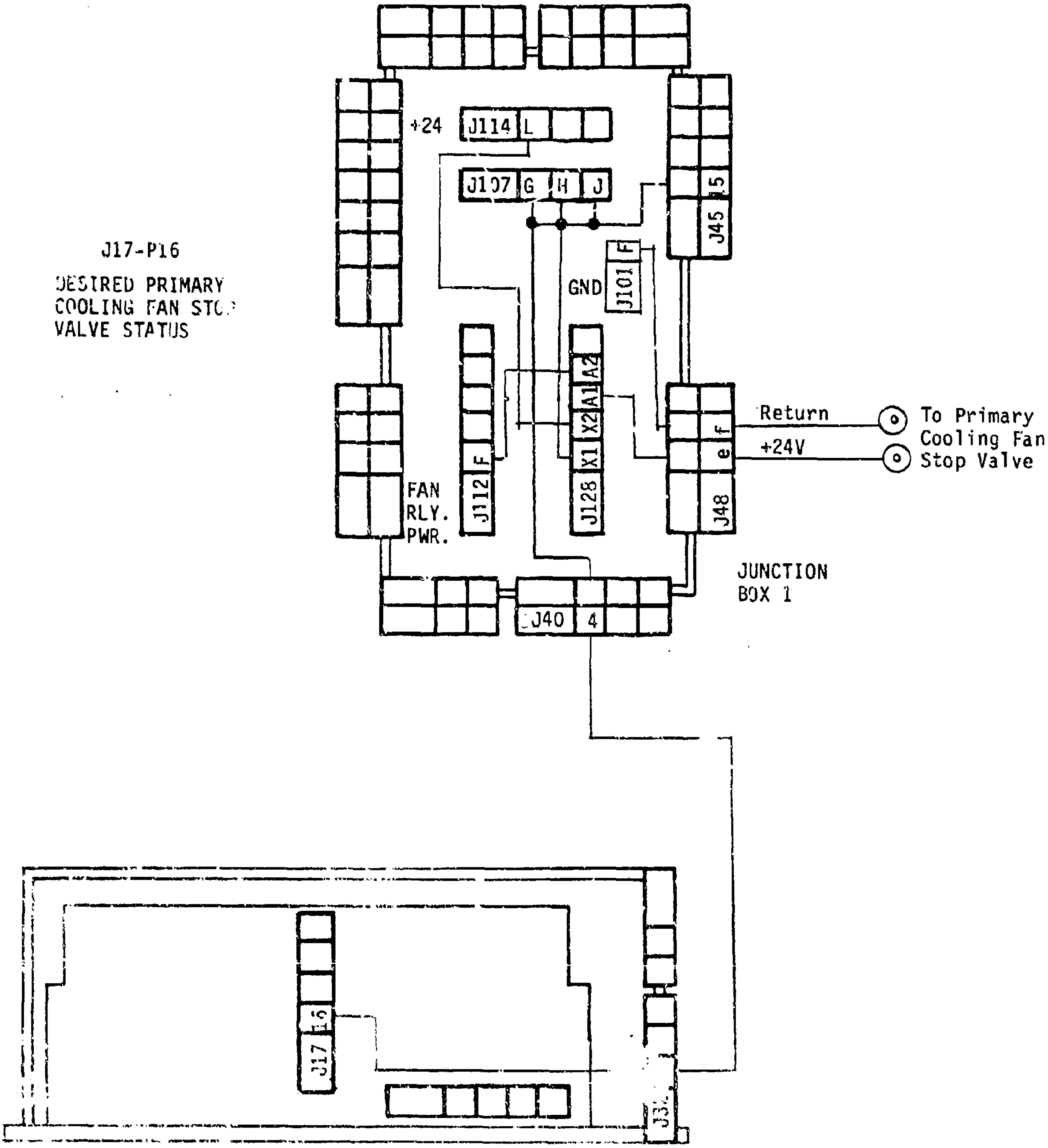
J17-P15

DESIRED SECONDARY GRILL OPEN VALVE STATUS



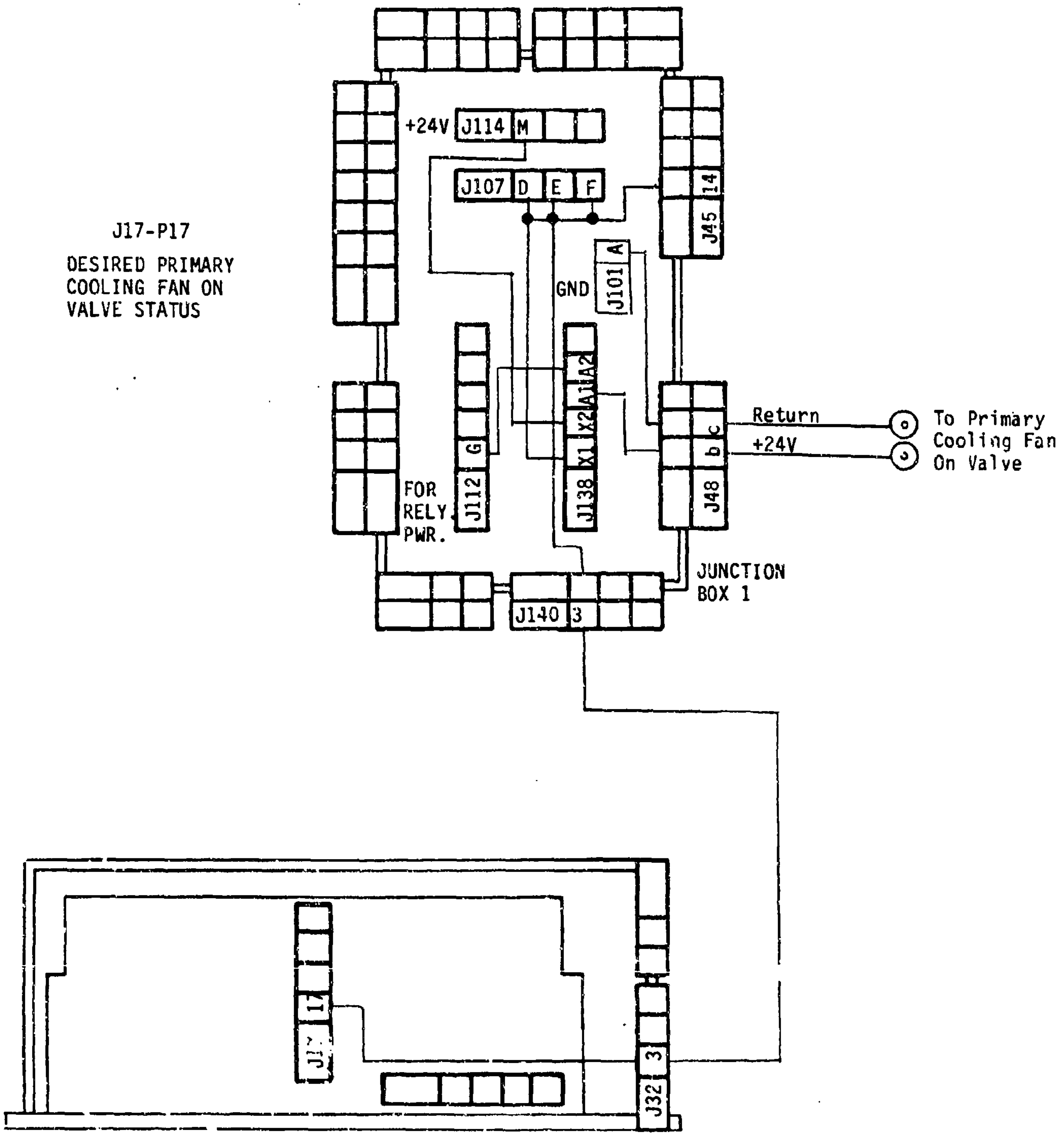
J17-P16

DESIRED PRIMARY COOLING FAN STOP VALVE STATUS



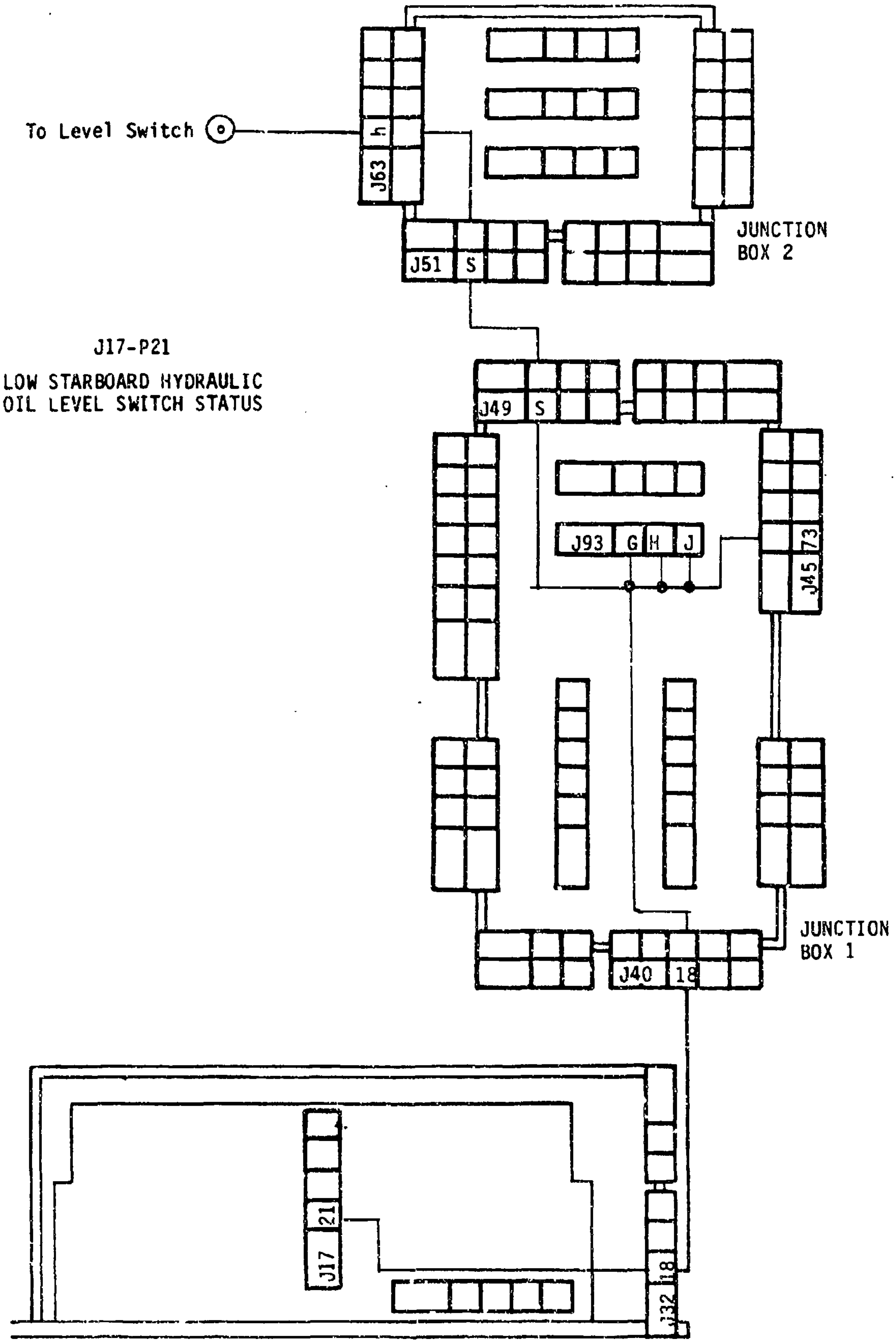
J17-P17

DESIRED PRIMARY COOLING FAN ON VALVE STATUS



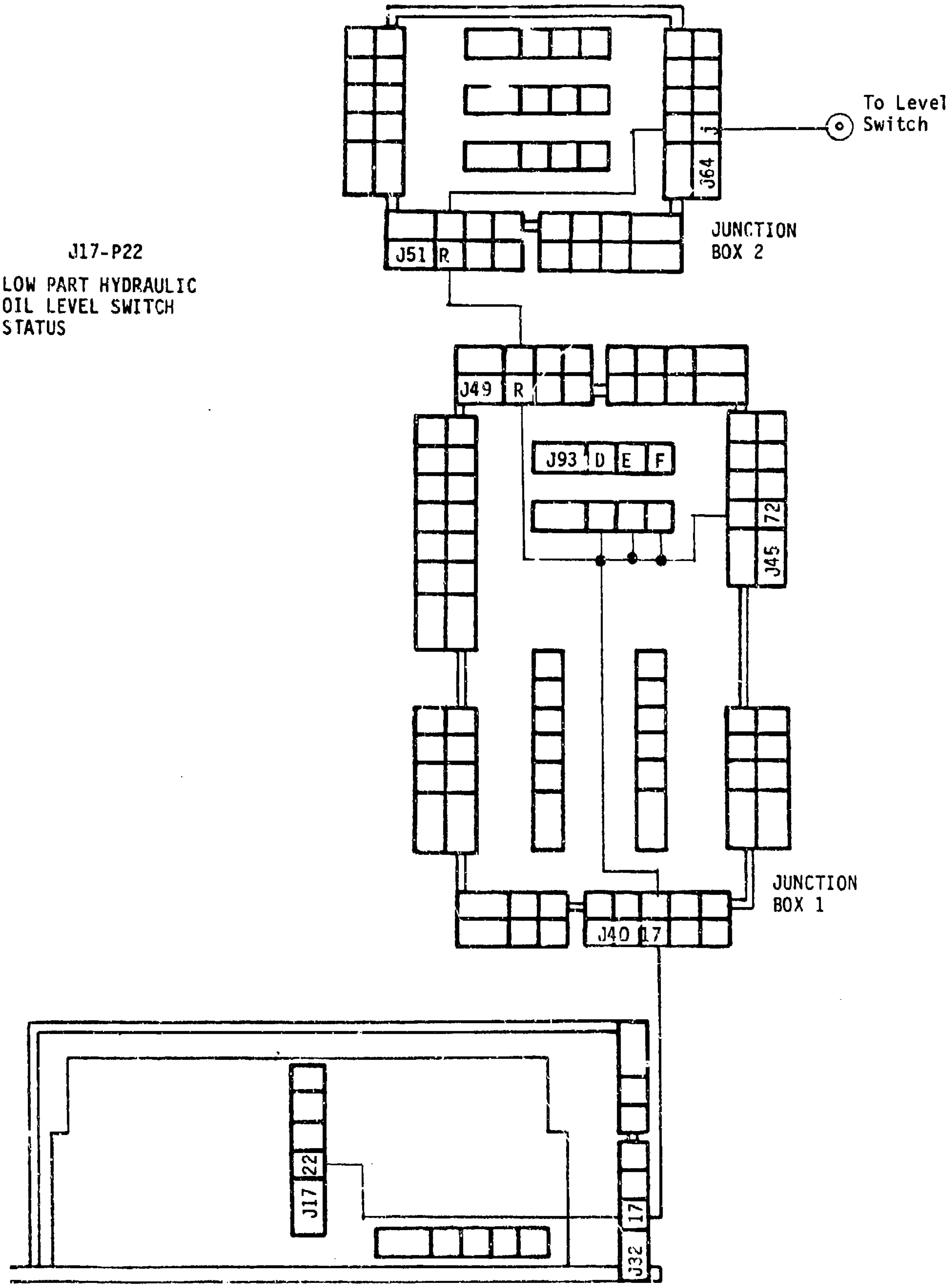
J17-P21

LOW STARBOARD HYDRAULIC
OIL LEVEL SWITCH STATUS



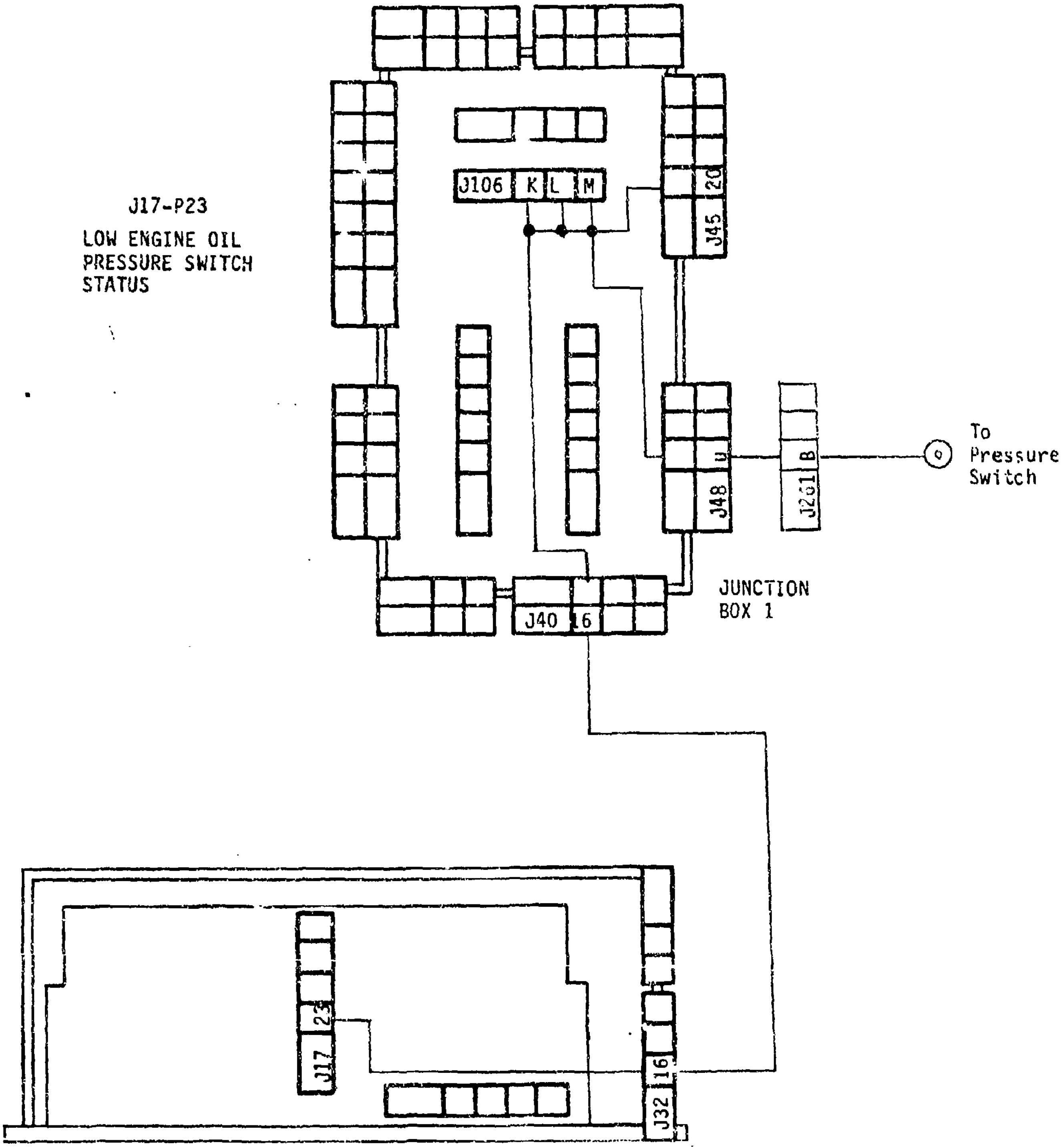
J17-P22

LOW PART HYDRAULIC
OIL LEVEL SWITCH
STATUS



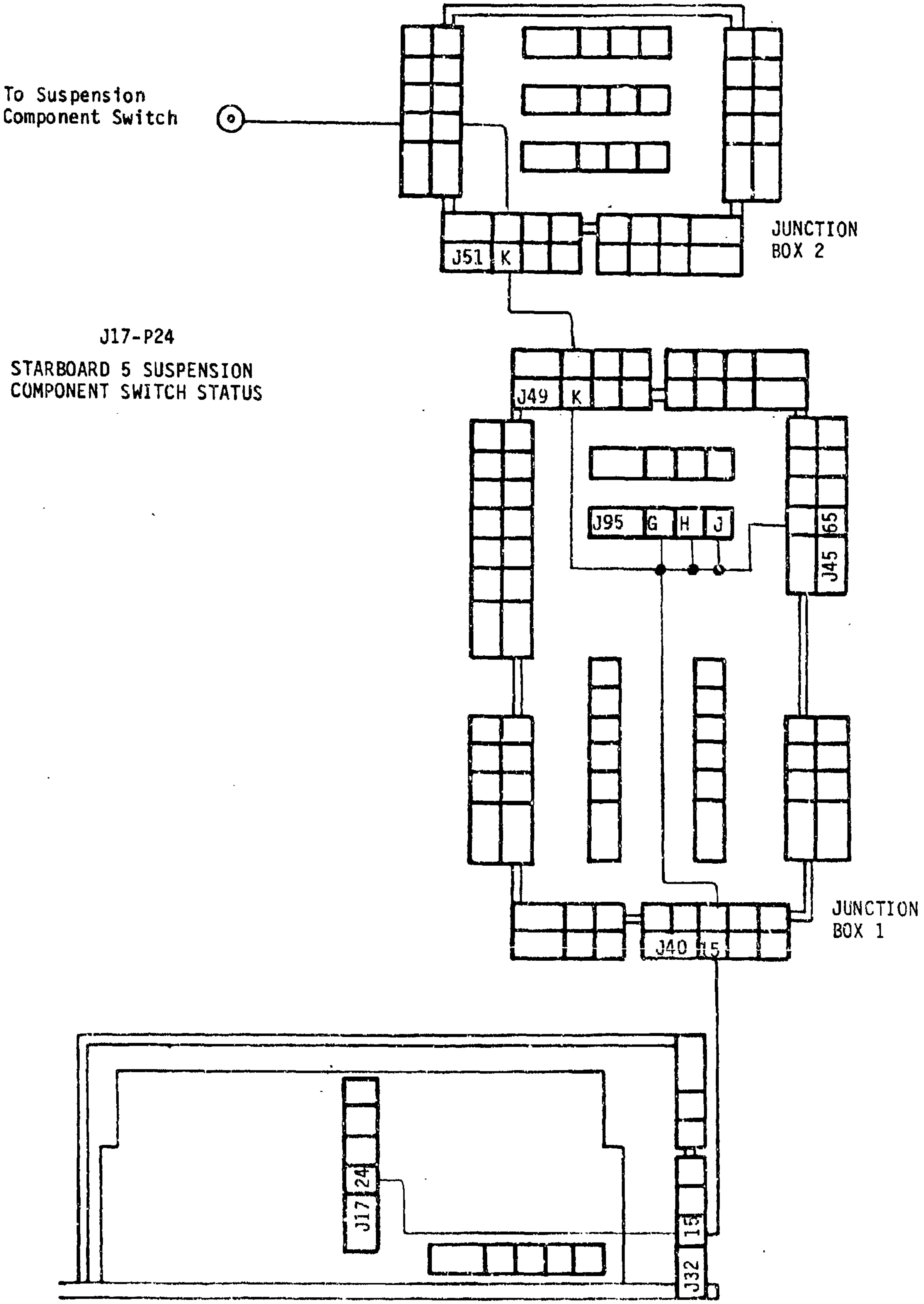
J17-P23

LOW ENGINE OIL PRESSURE SWITCH STATUS



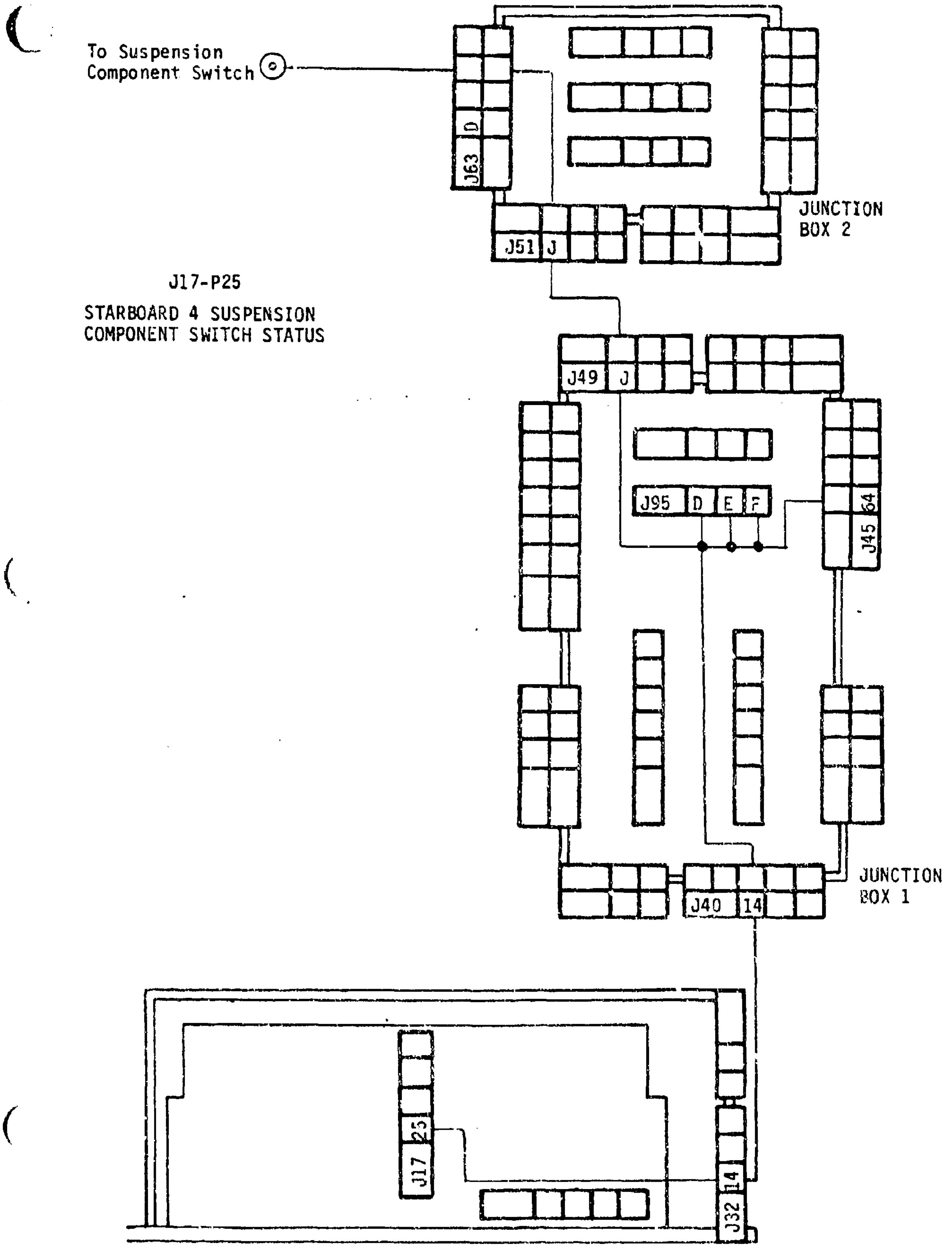
J17-P24

STARBOARD 5 SUSPENSION
COMPONENT SWITCH STATUS



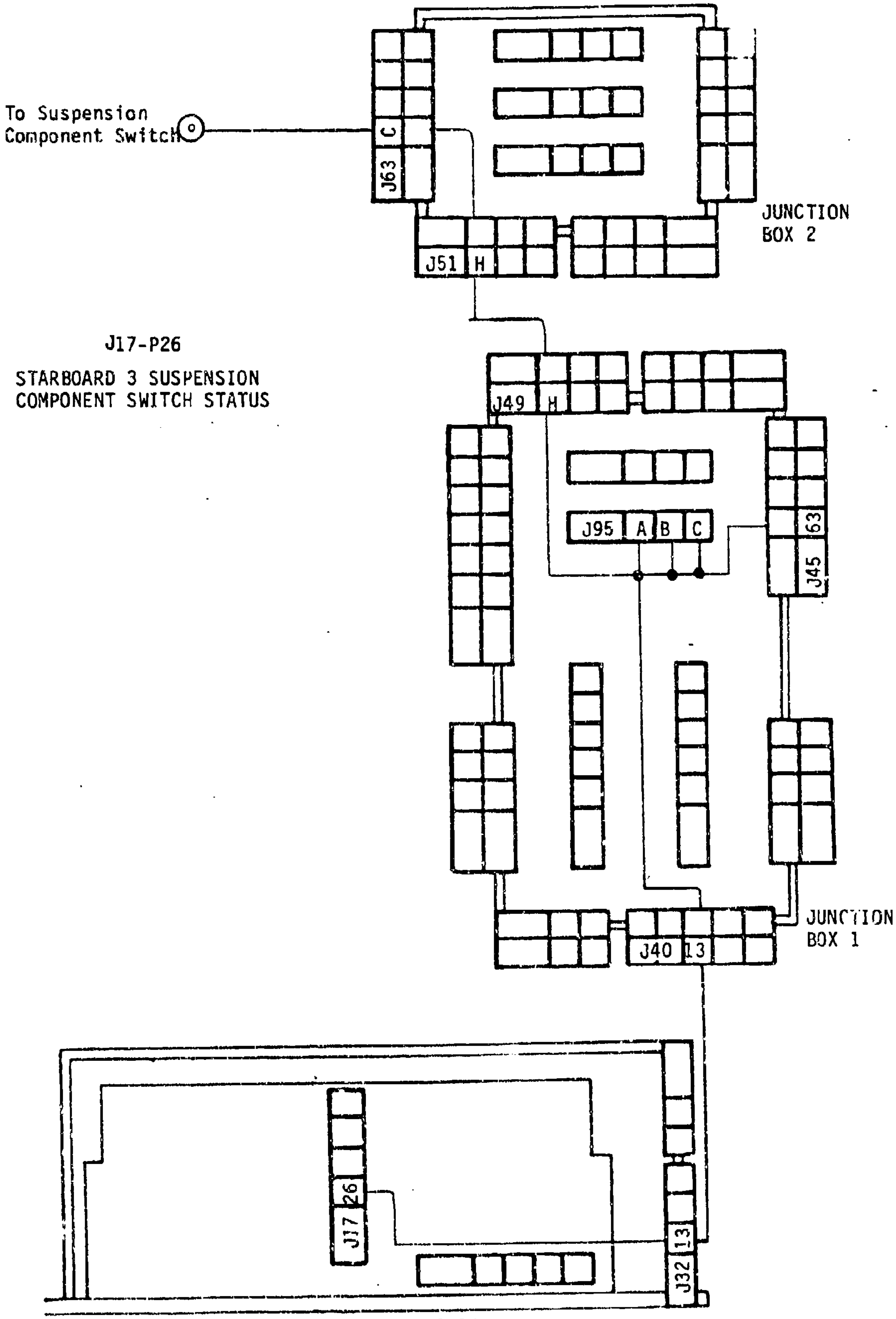
J17-P25

STARBOARD 4 SUSPENSION
COMPONENT SWITCH STATUS



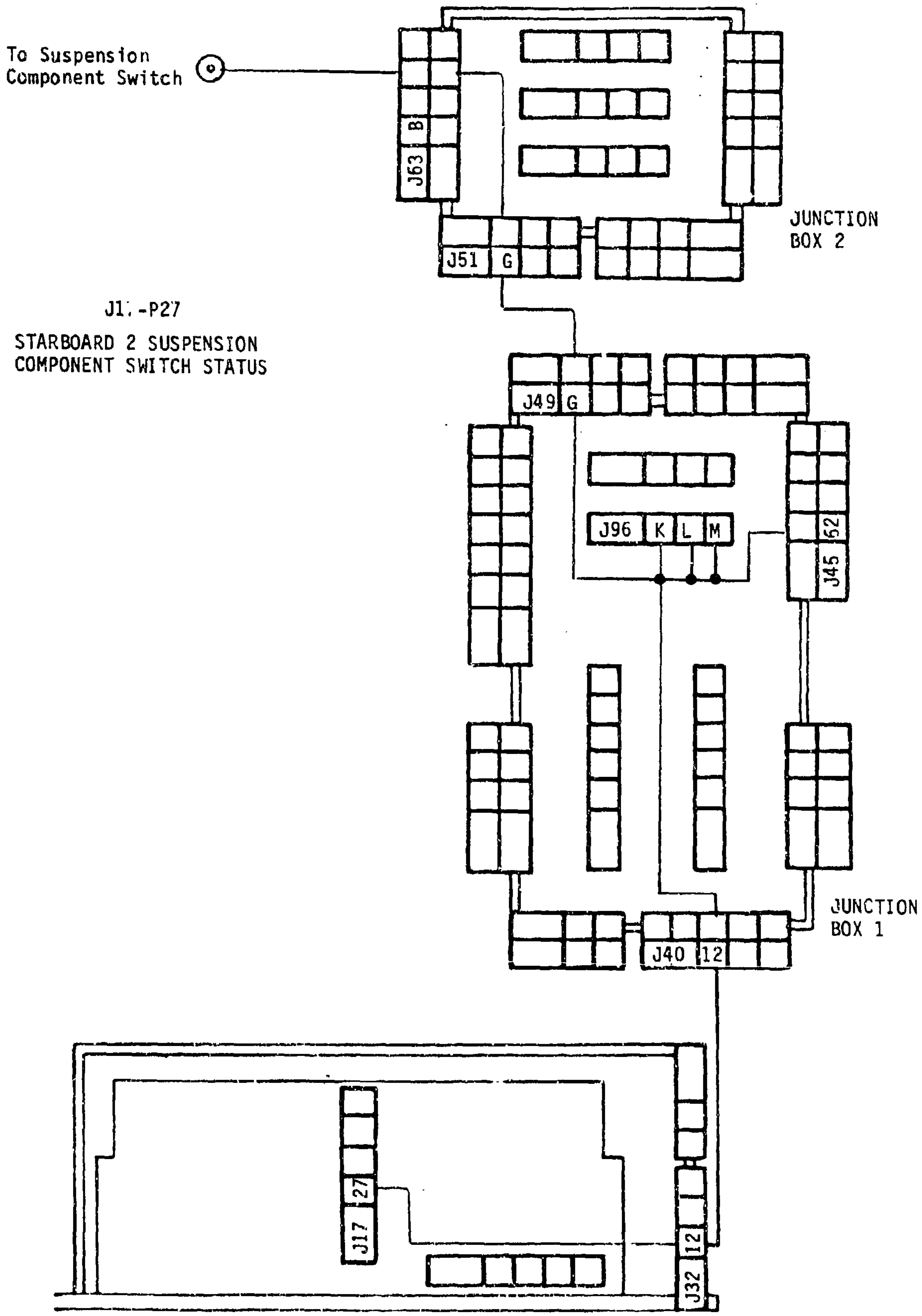
J17-P26

STARBOARD 3 SUSPENSION
COMPONENT SWITCH STATUS



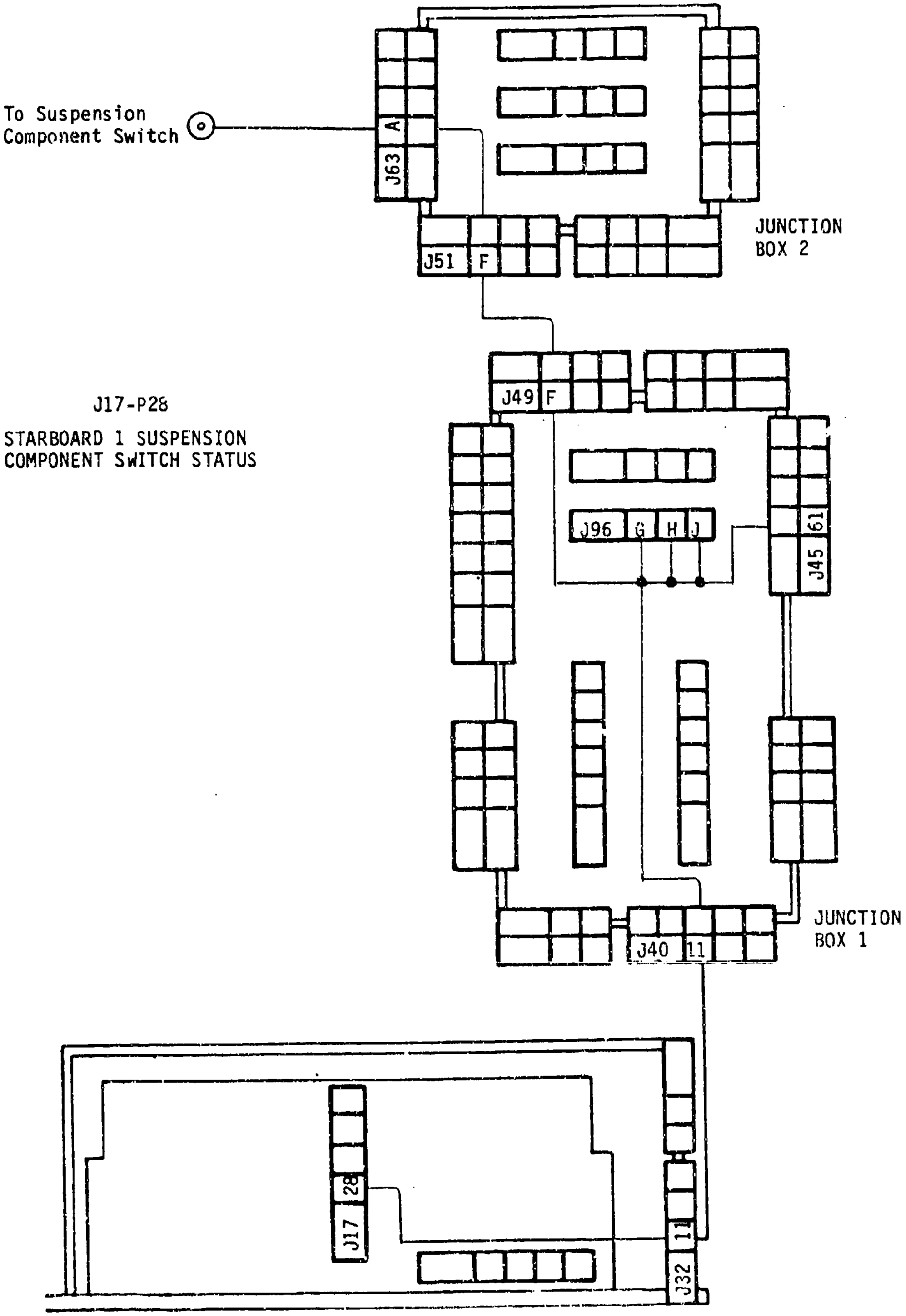
J17-P27

STARBOARD 2 SUSPENSION
COMPONENT SWITCH STATUS

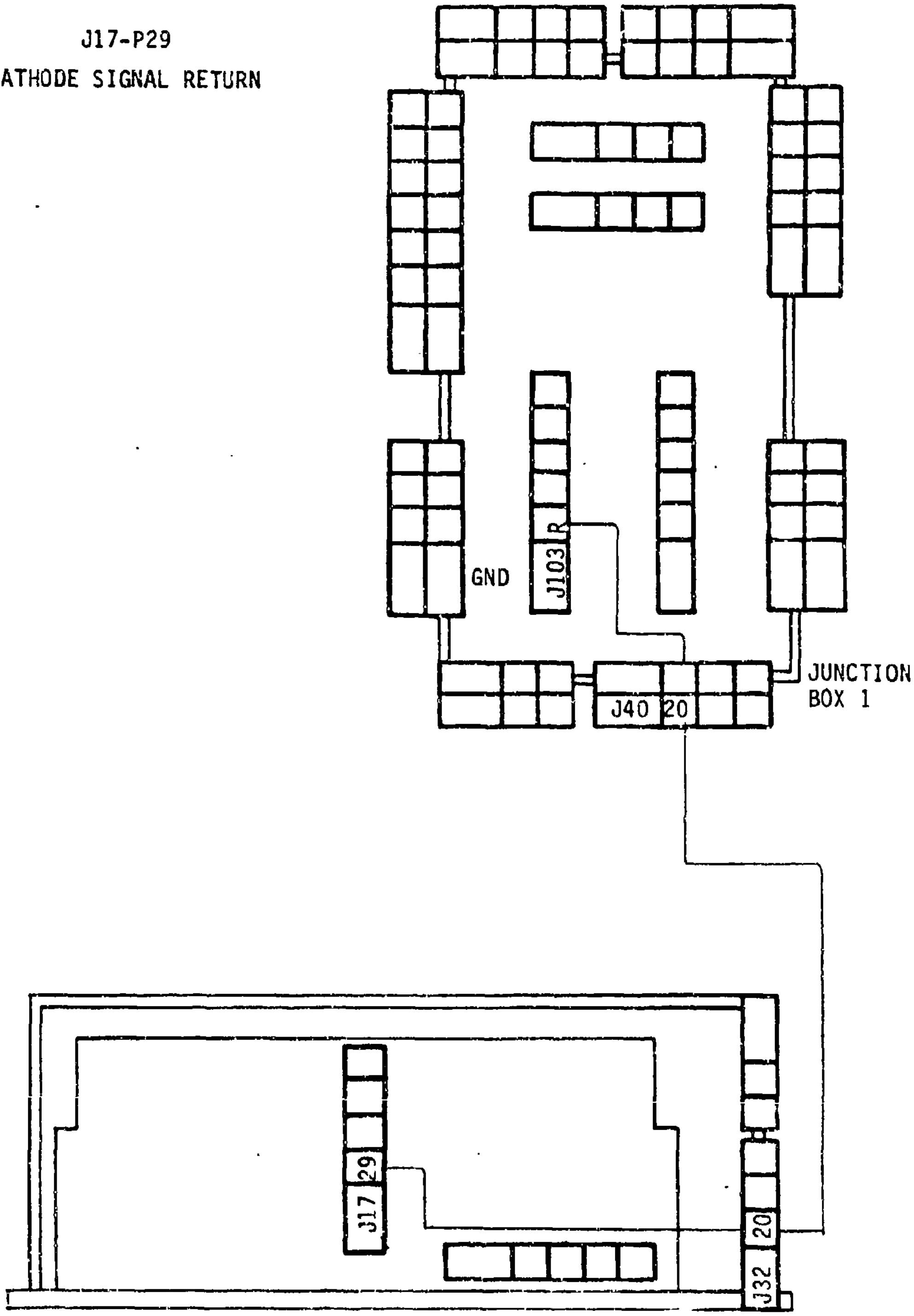


J17-P28

STARBOARD 1 SUSPENSION COMPONENT SWITCH STATUS

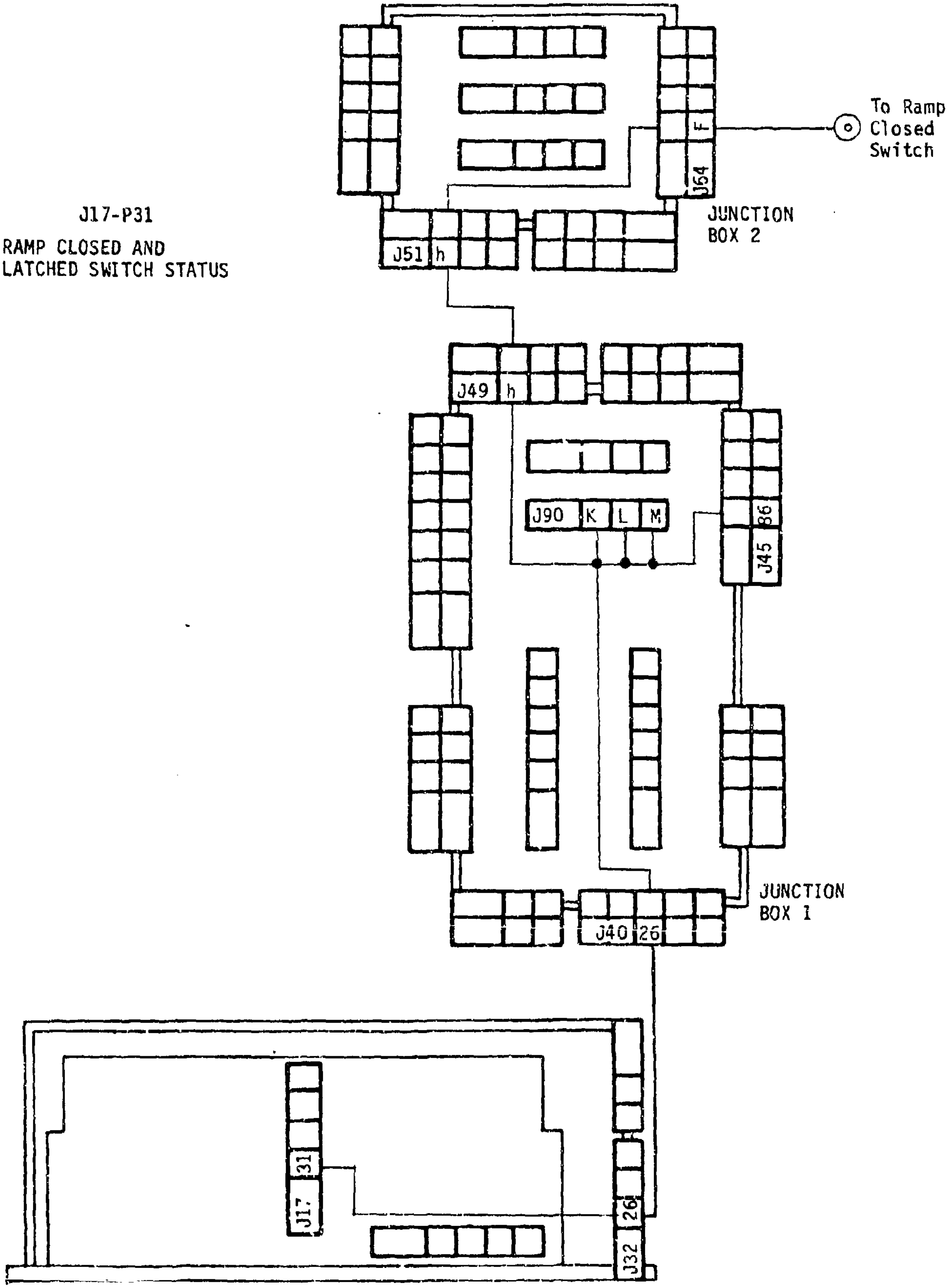


J17-P29
CATHODE SIGNAL RETURN



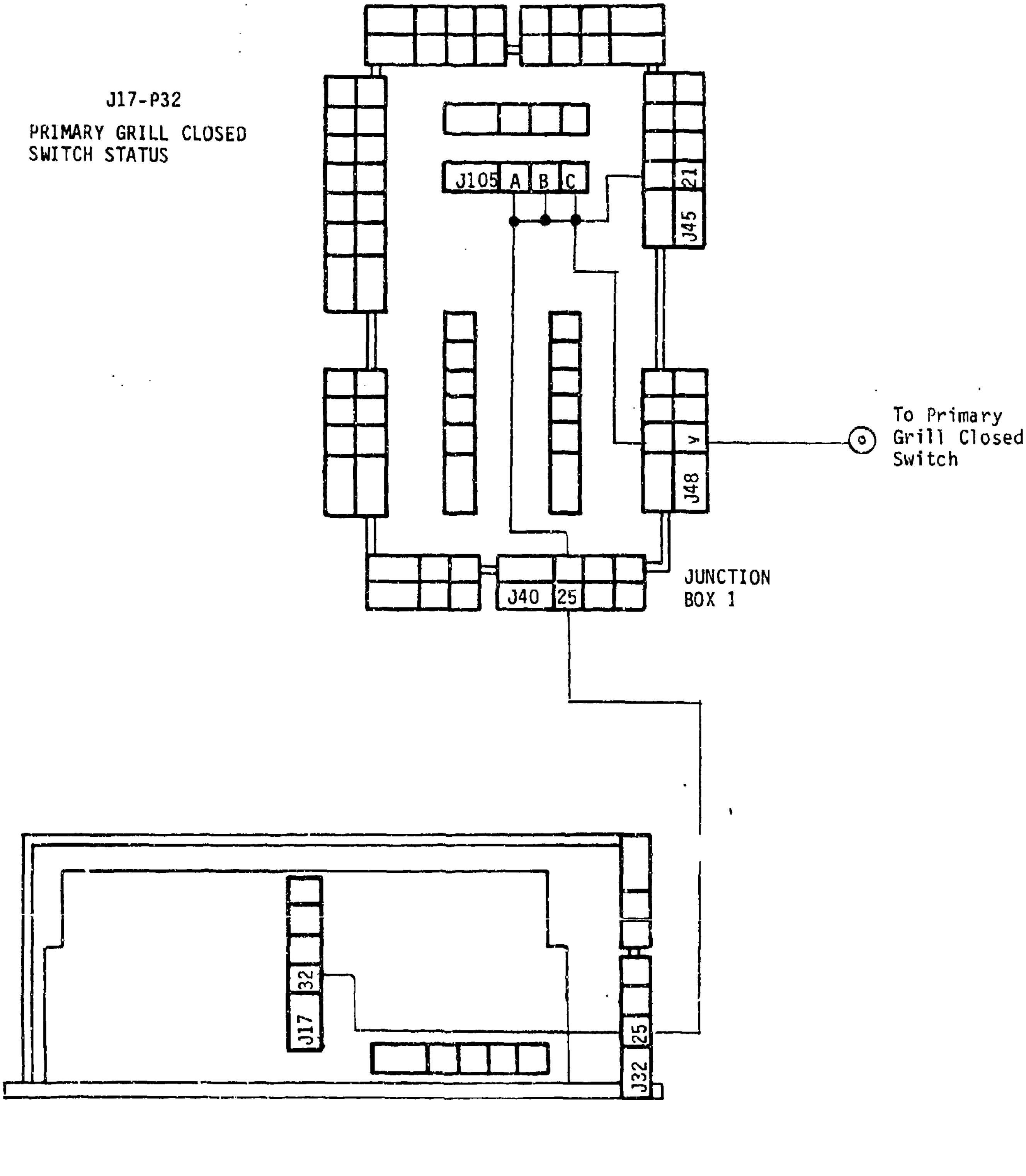
J17-P31

RAMP CLOSED AND
LATCHED SWITCH STATUS



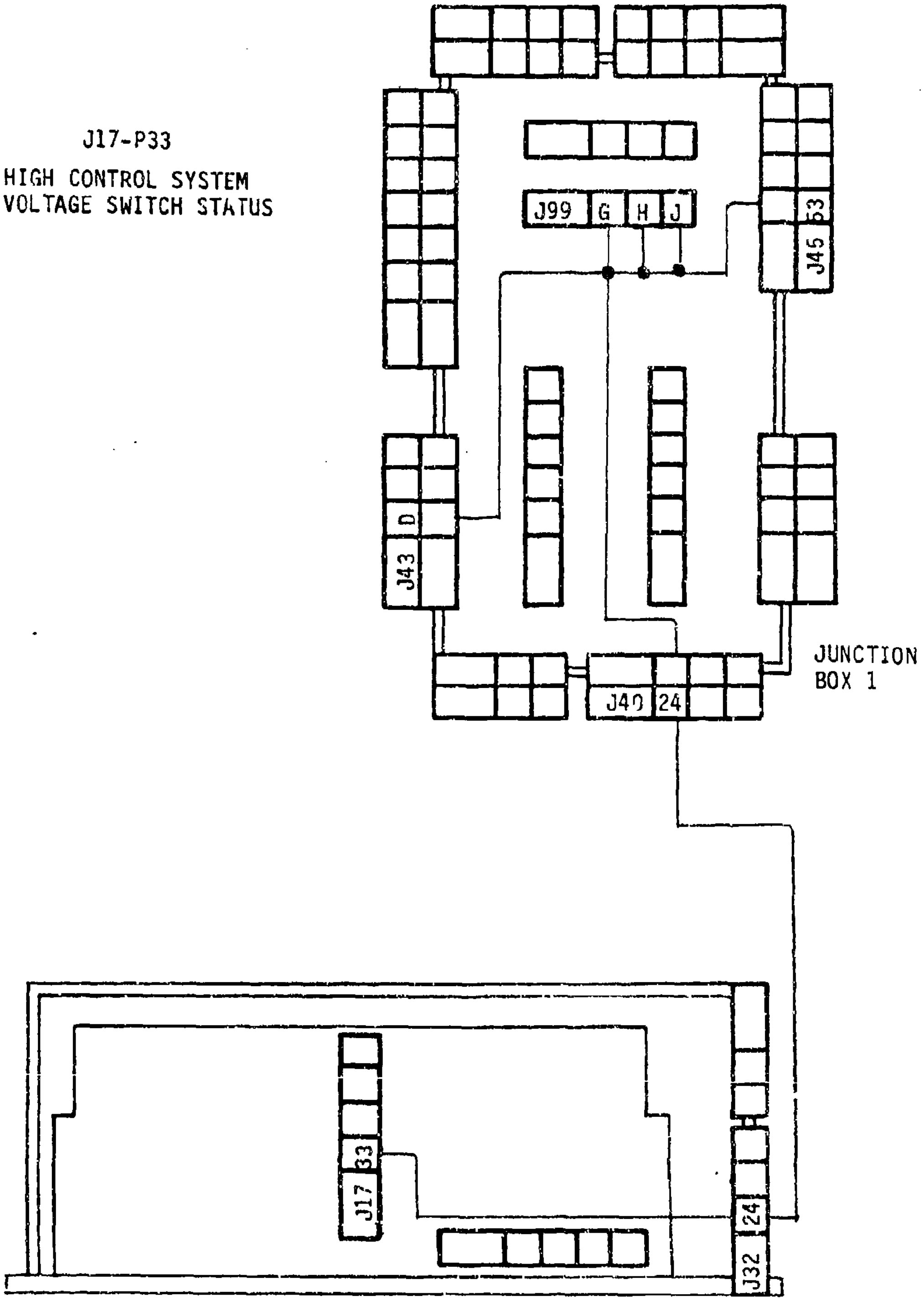
J17-P32

PRIMARY GRILL CLOSED
SWITCH STATUS



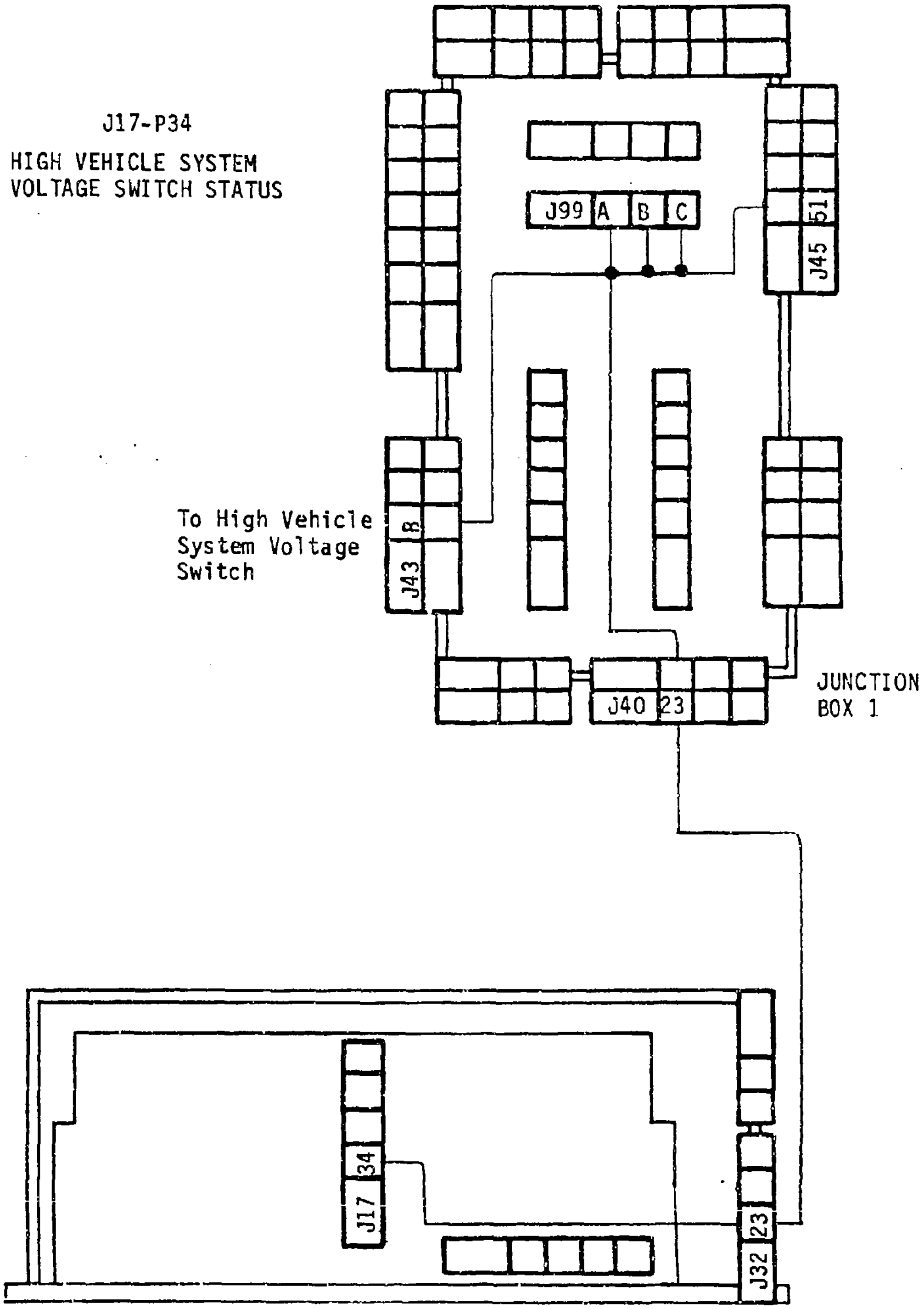
J17-P33

HIGH CONTROL SYSTEM
VOLTAGE SWITCH STATUS



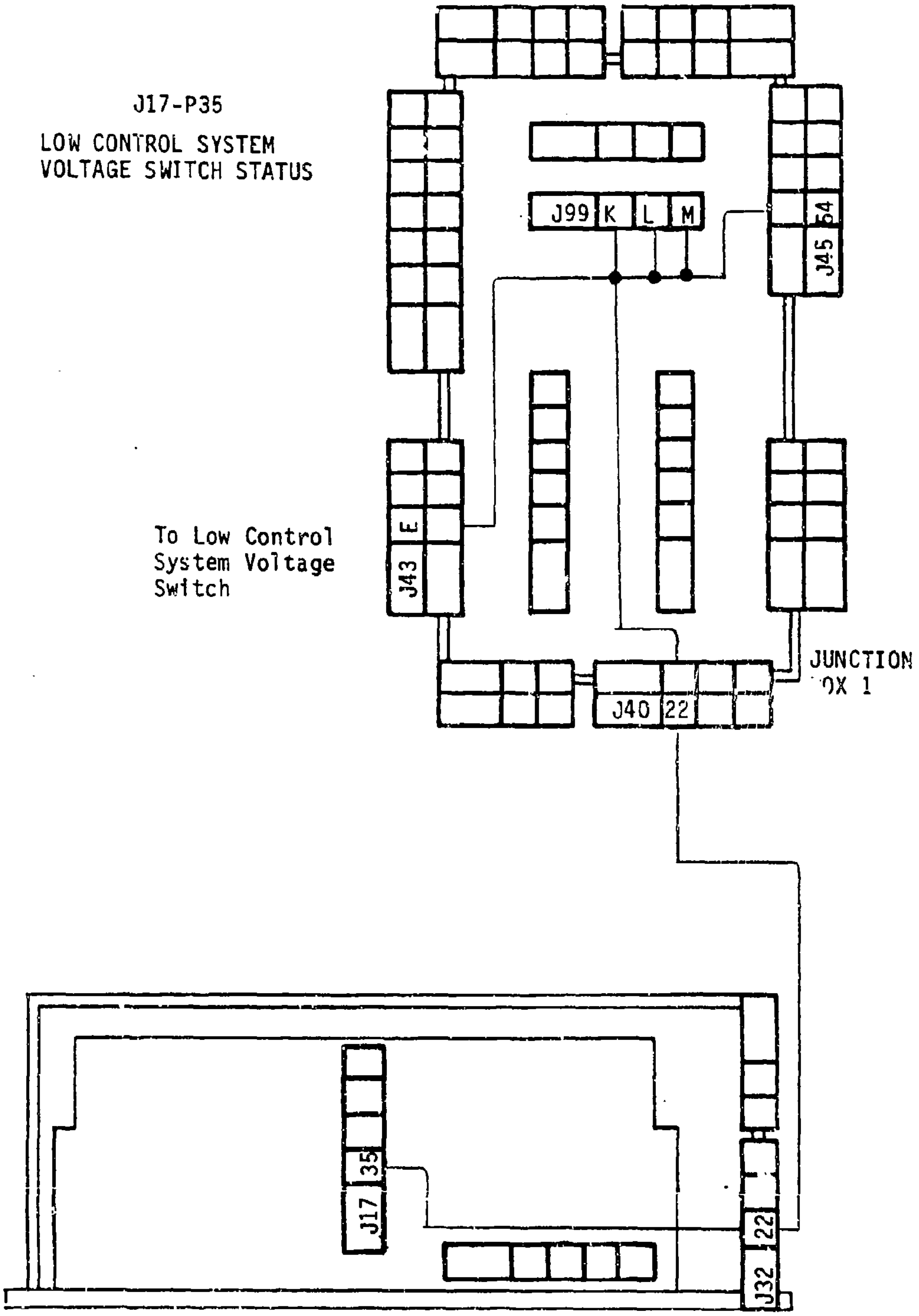
J17-P34

HIGH VEHICLE SYSTEM VOLTAGE SWITCH STATUS



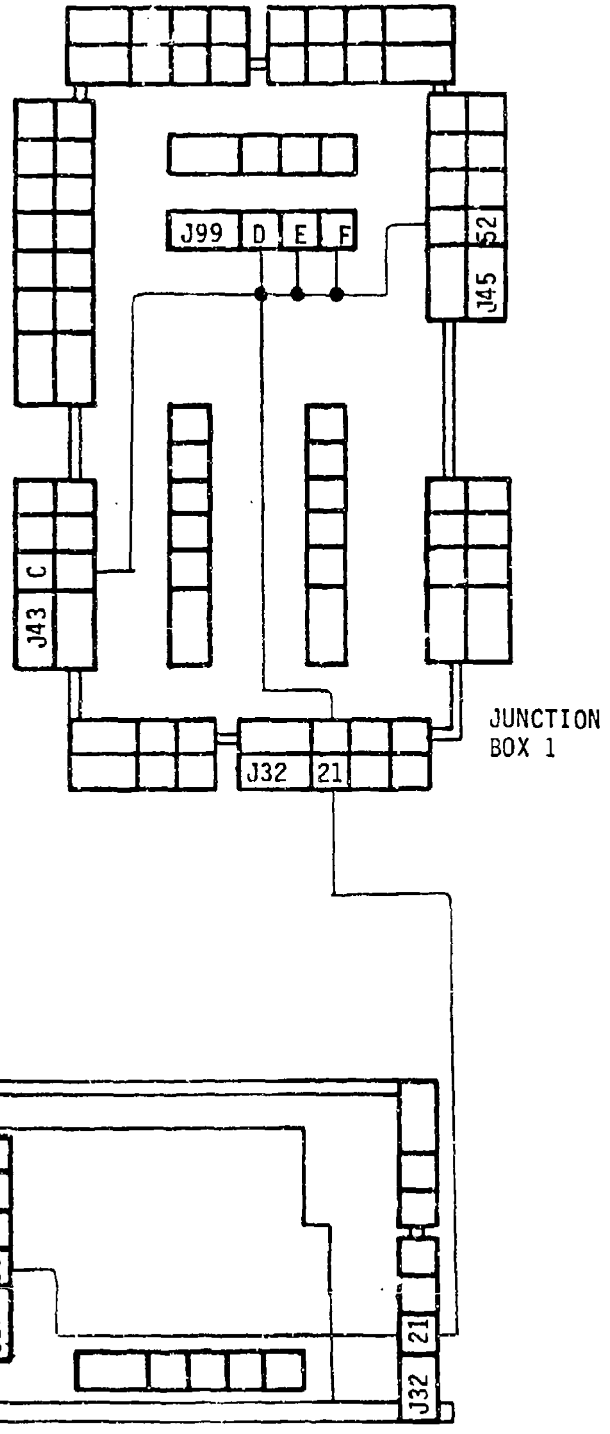
J17-P35

LOW CONTROL SYSTEM VOLTAGE SWITCH STATUS



J17-P36

LOW VEHICLE SYSTEM VOLTAGE SWITCH STATUS

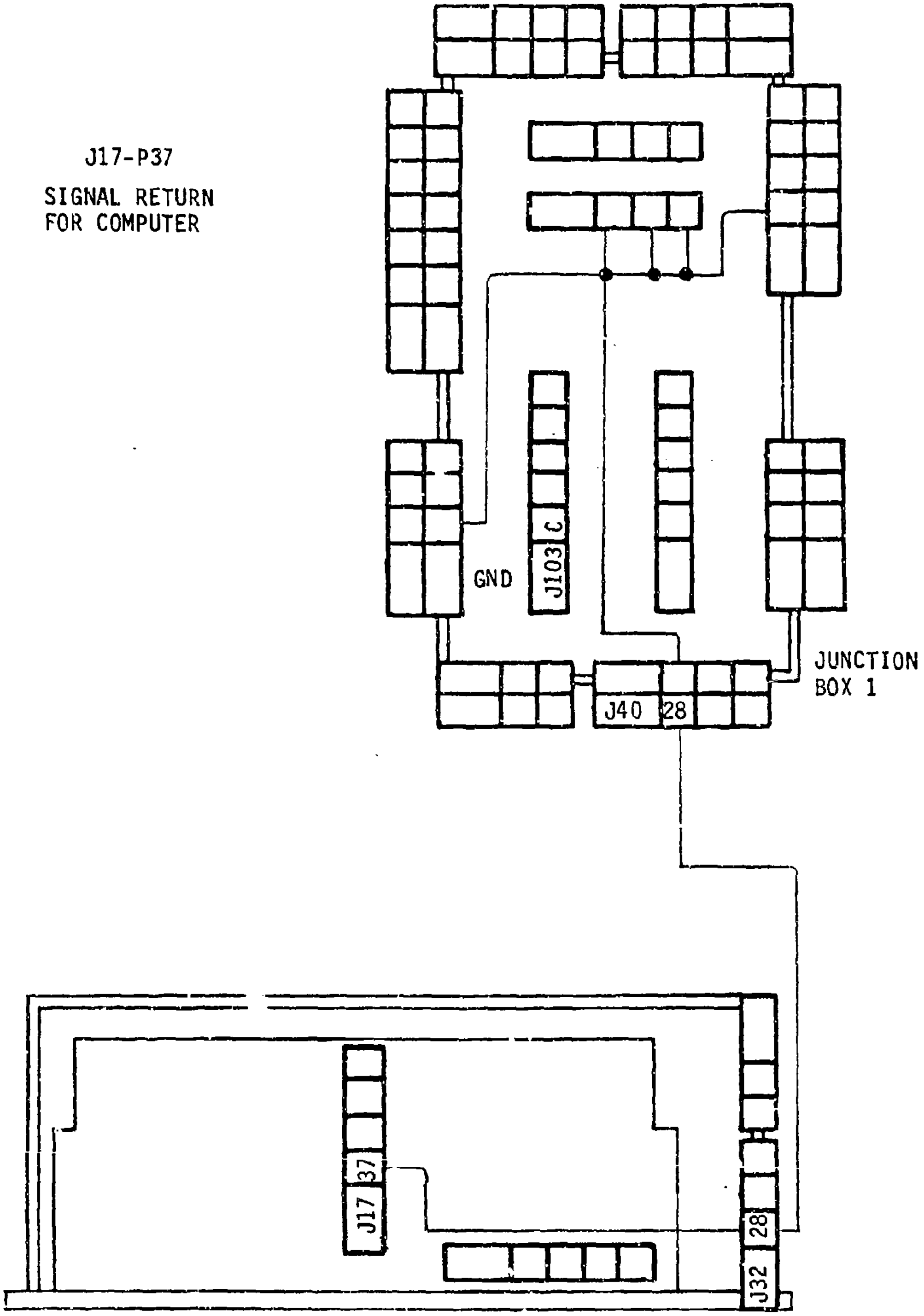


To Low Vehicle System Voltage Switch

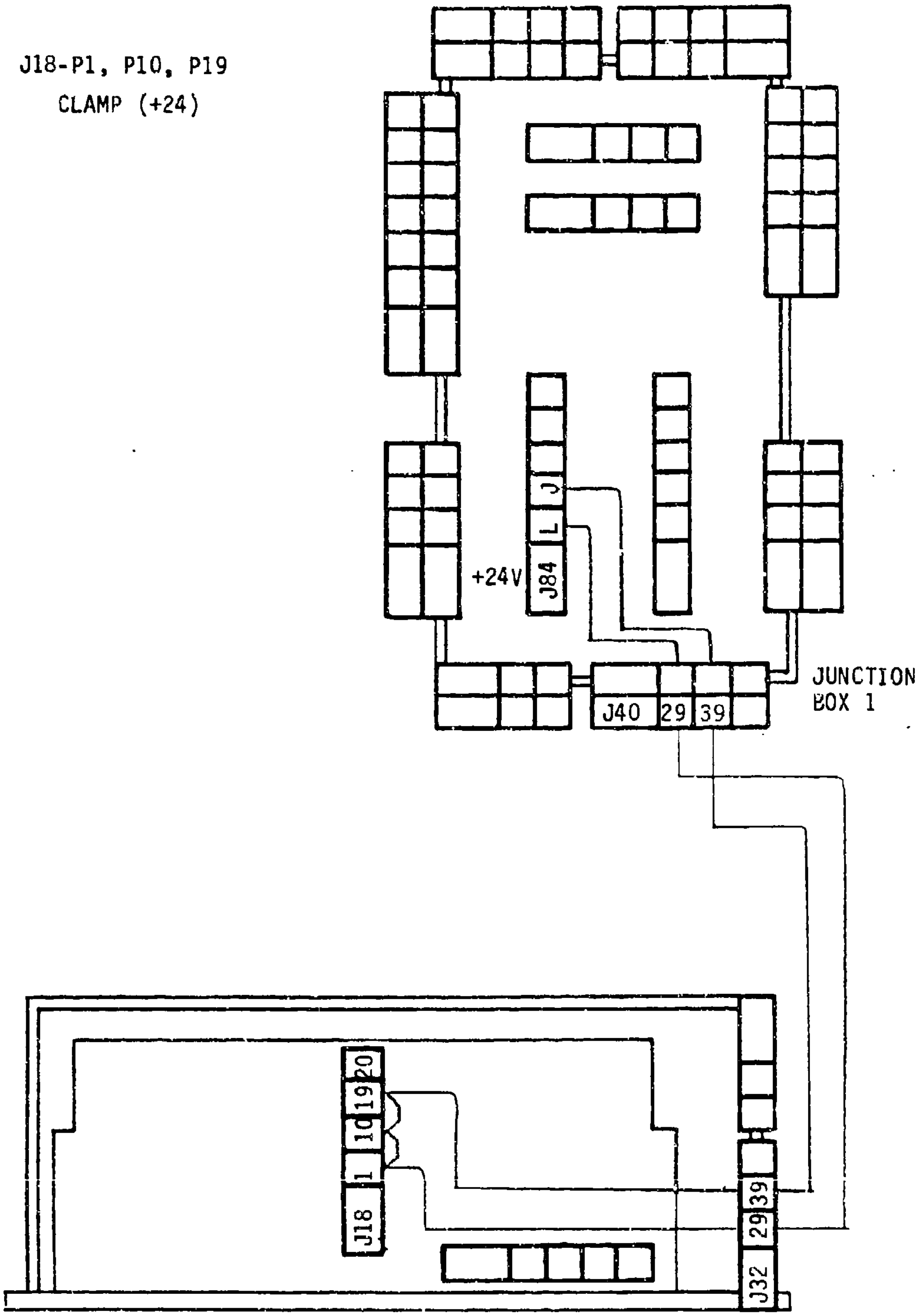
JUNCTION BOX 1

J17-P37

SIGNAL RETURN FOR COMPUTER

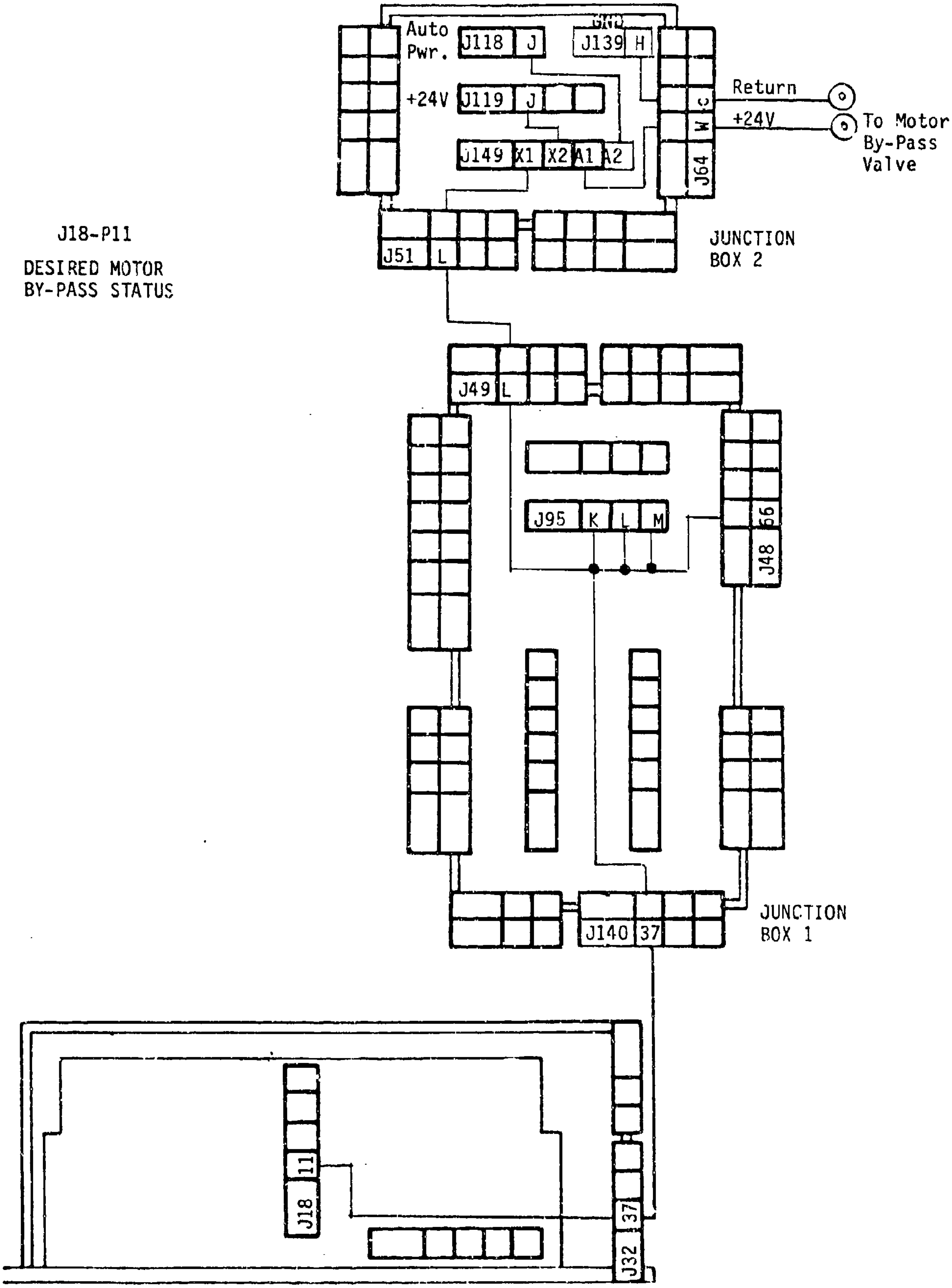


J18-P1, P10, P19
CLAMP (+24)



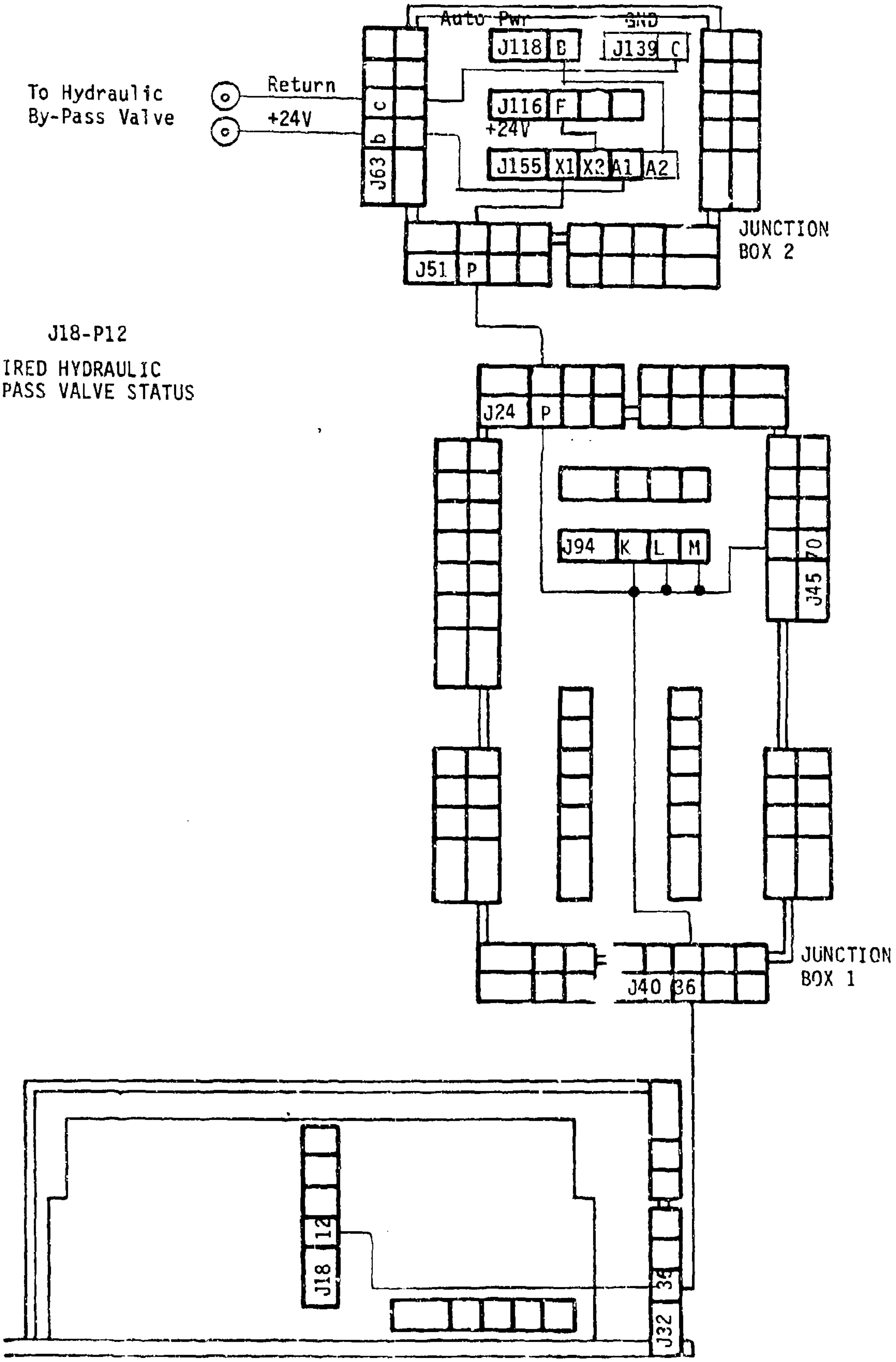
J18-P11

DESIRED MOTOR
BY-PASS STATUS



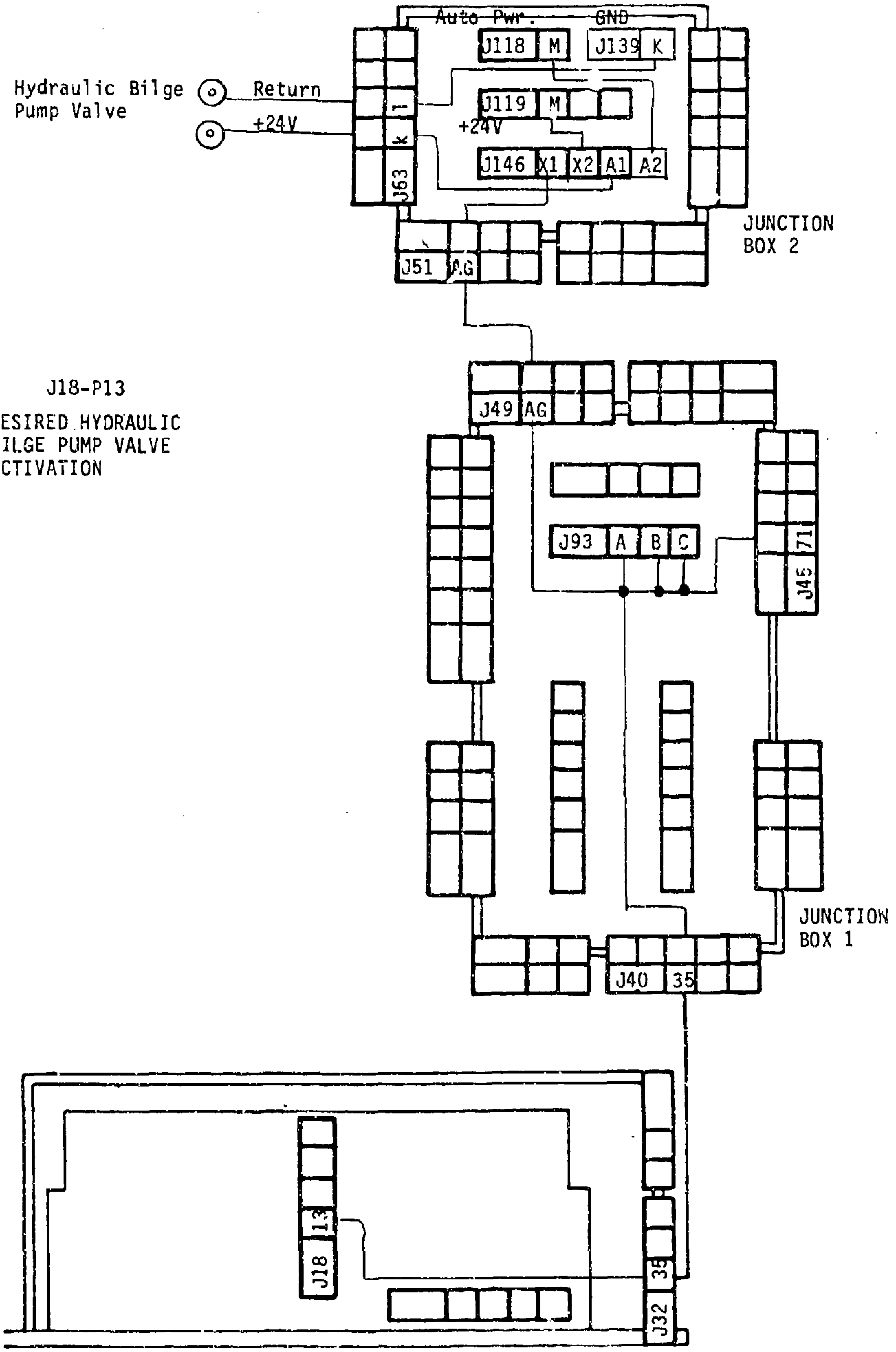
J18-P12

DESIRED HYDRAULIC
BY-PASS VALVE STATUS



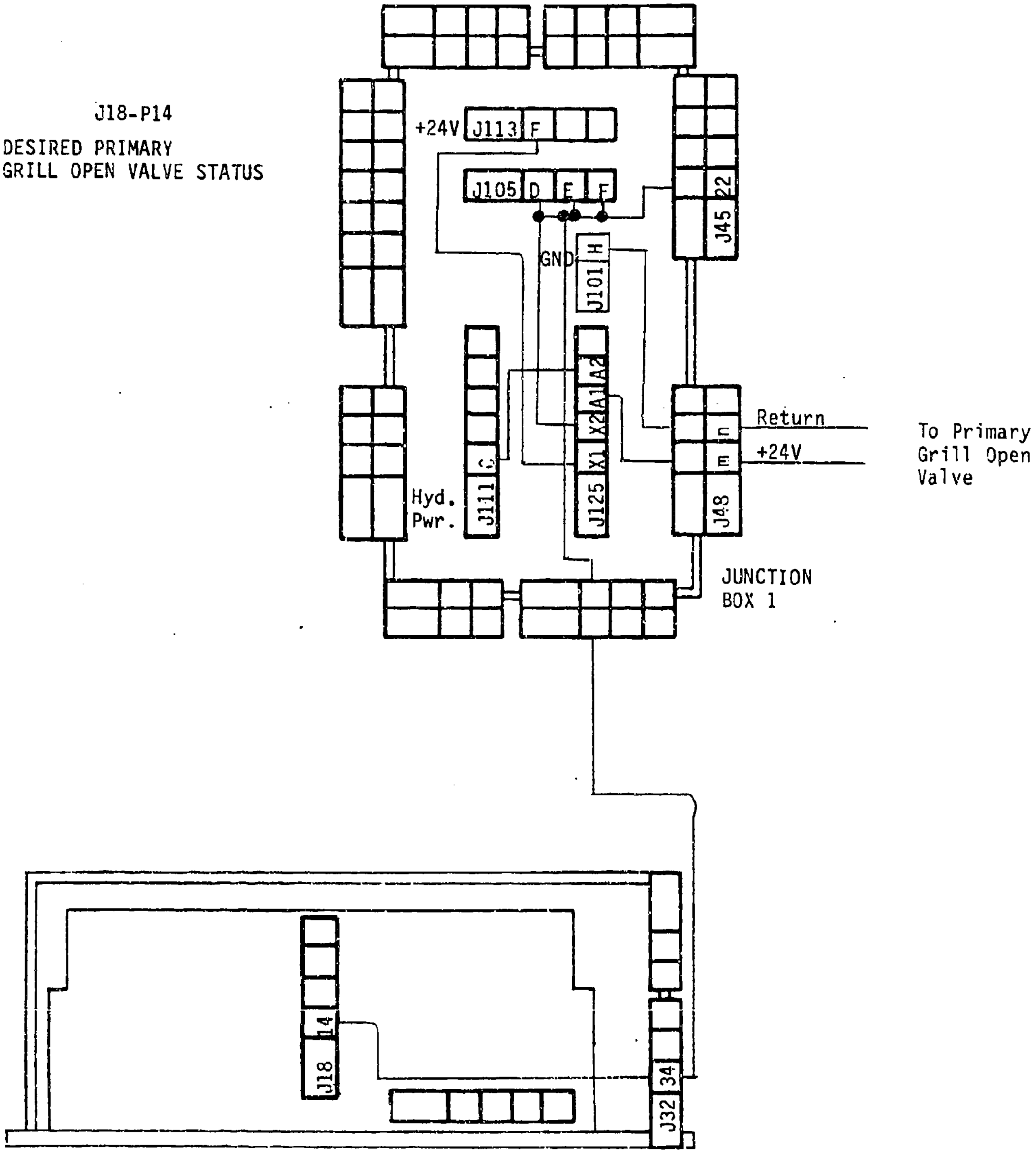
J18-P13

DESIRED HYDRAULIC BILGE PUMP VALVE ACTIVATION

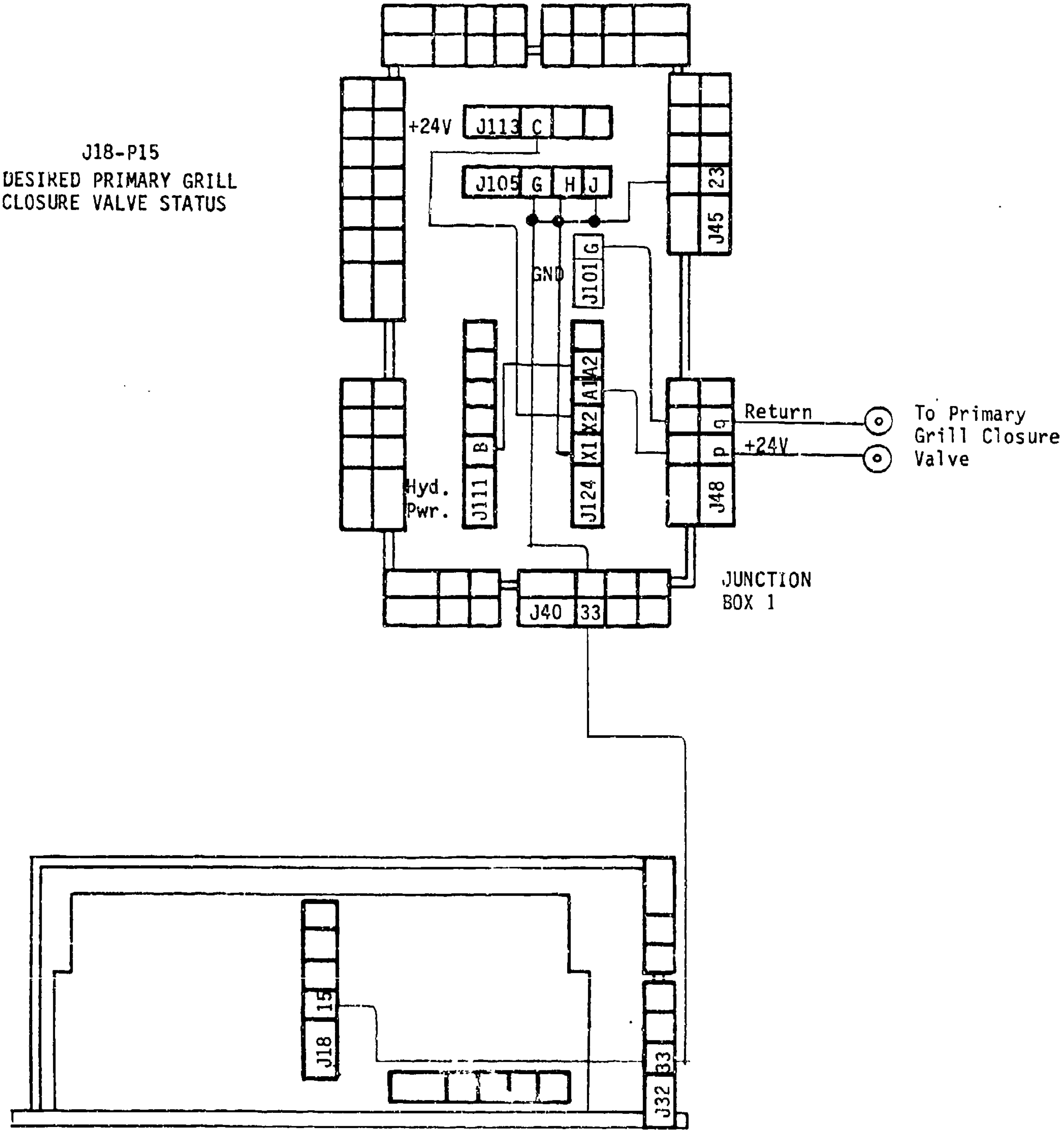


J18-P14

DESIRED PRIMARY
GRILL OPEN VALVE STATUS

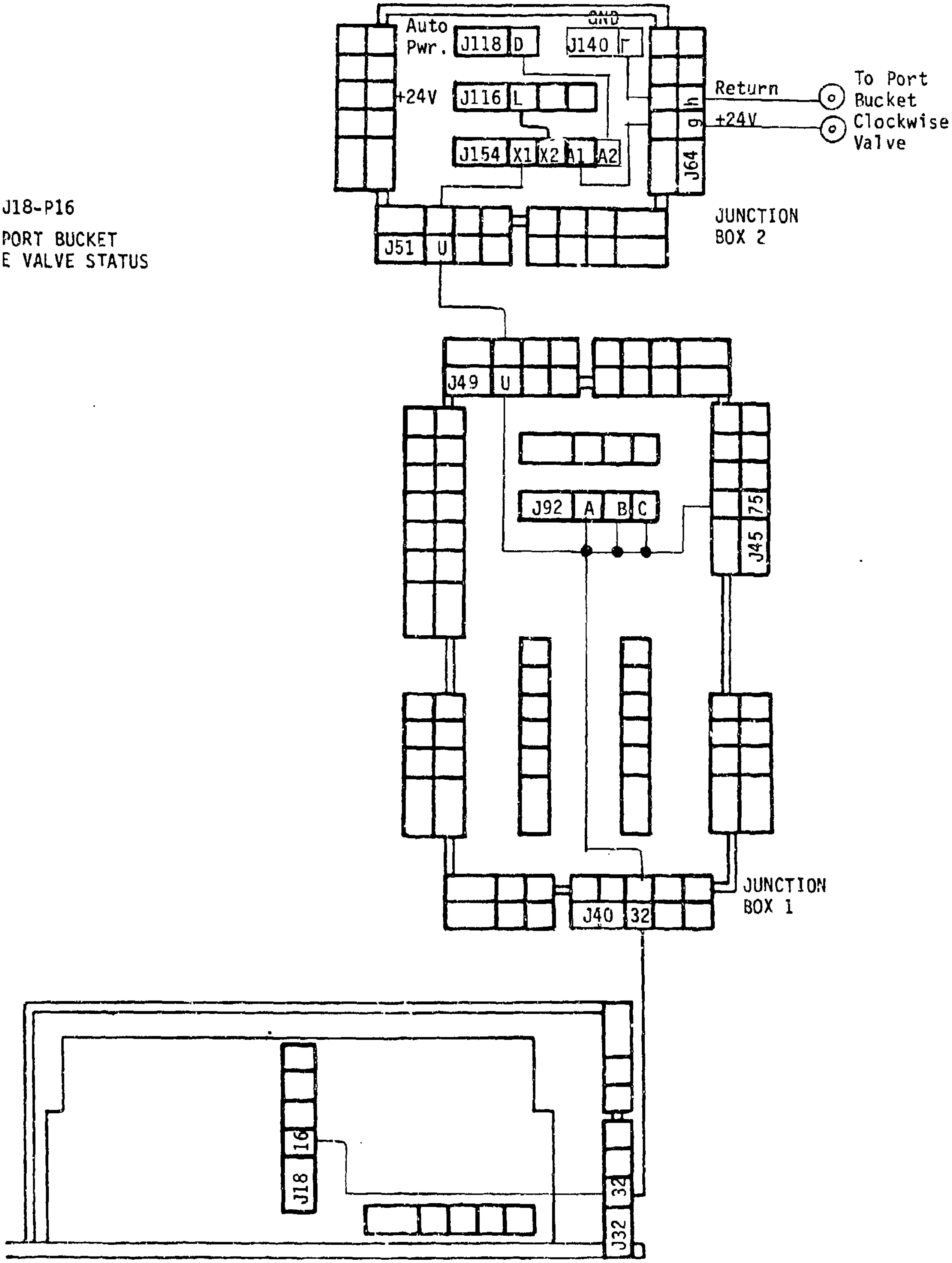


J18-P15
DESIRED PRIMARY GRILL
CLOSURE VALVE STATUS



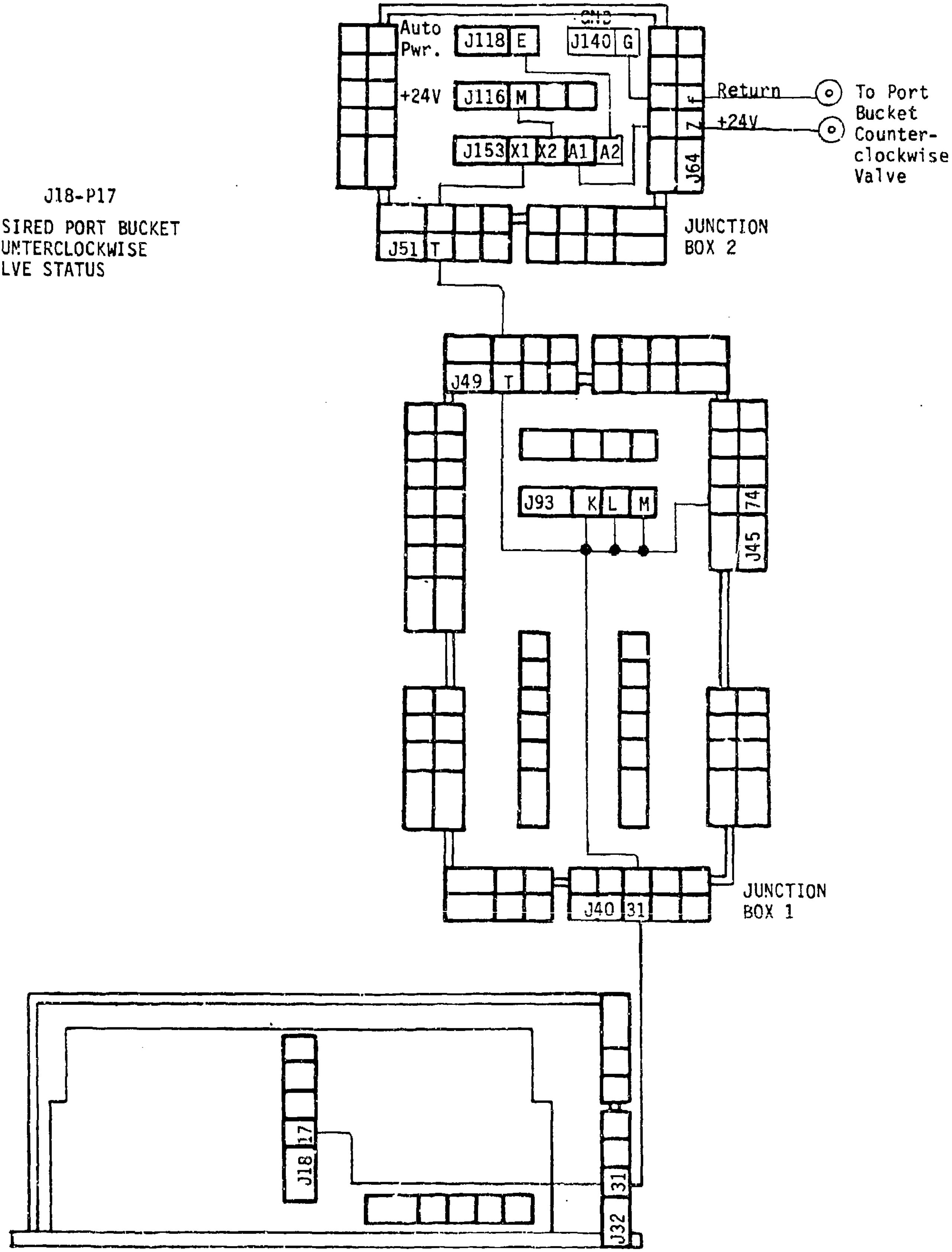
J18-P16

DESIRED PORT BUCKET CLOCKWISE VALVE STATUS



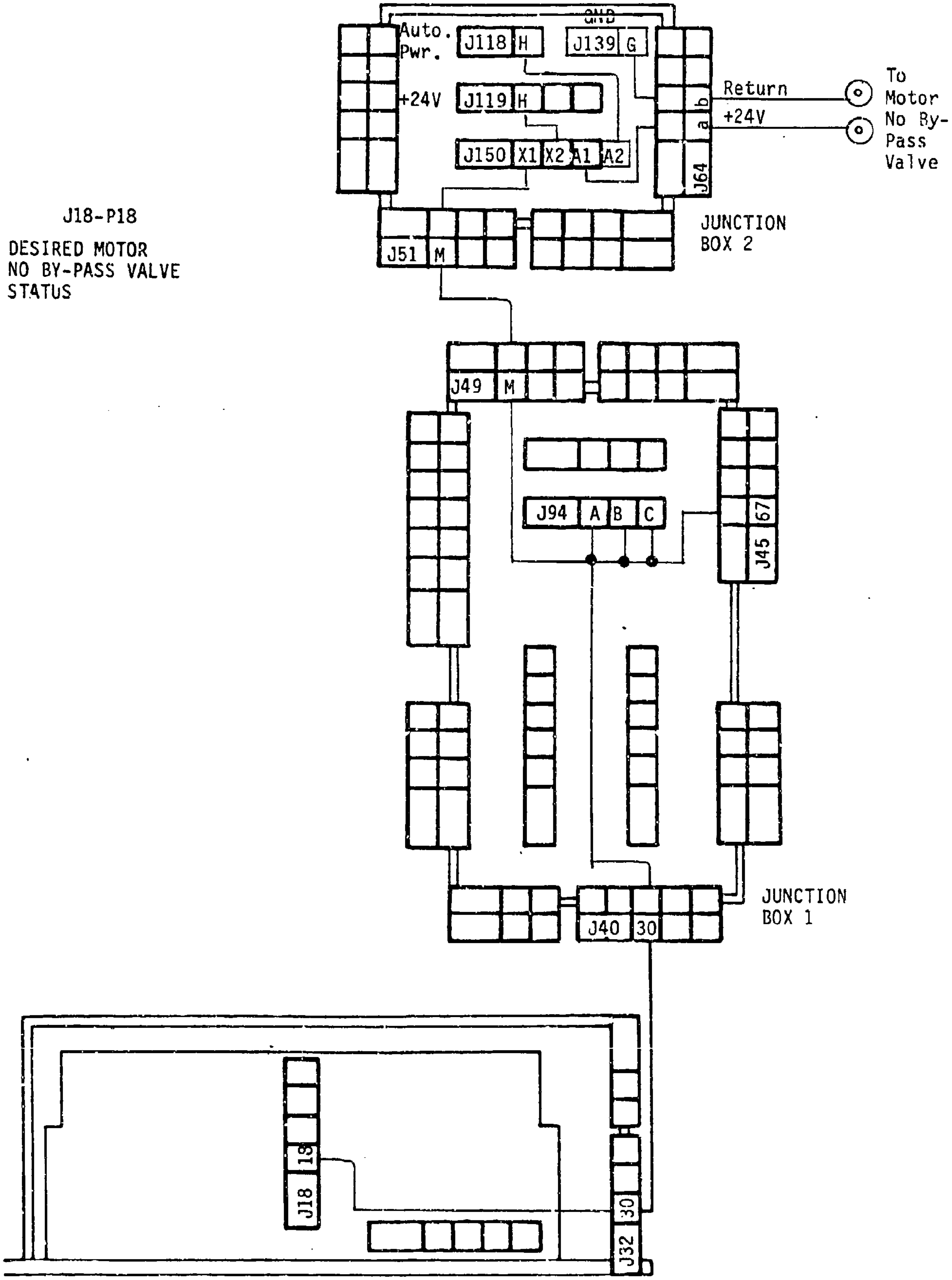
J18-P17

DESIRED PORT BUCKET COUNTERCLOCKWISE VALVE STATUS



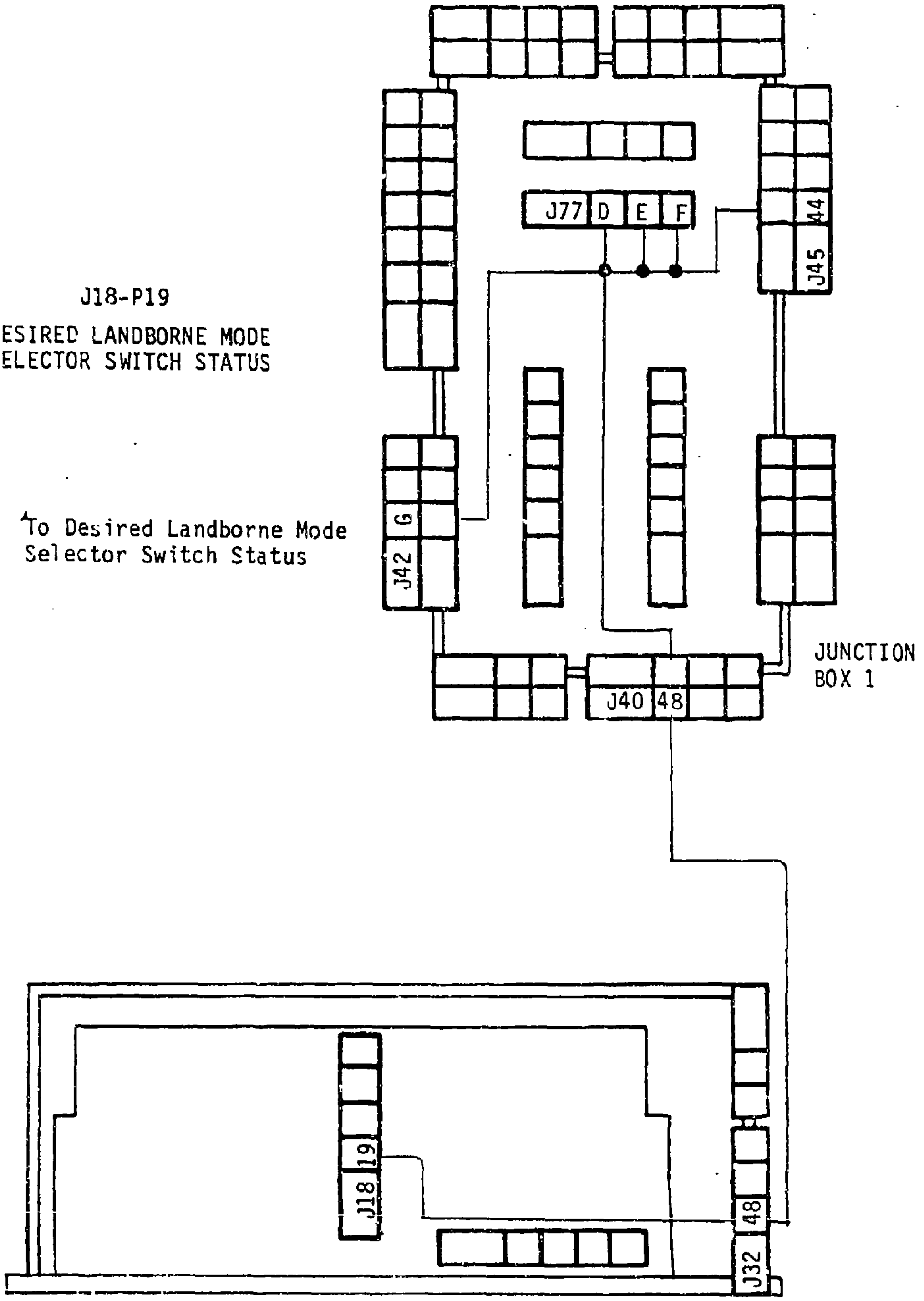
J18-P18

DESIRED MOTOR
NO BY-PASS VALVE
STATUS



J18-P19

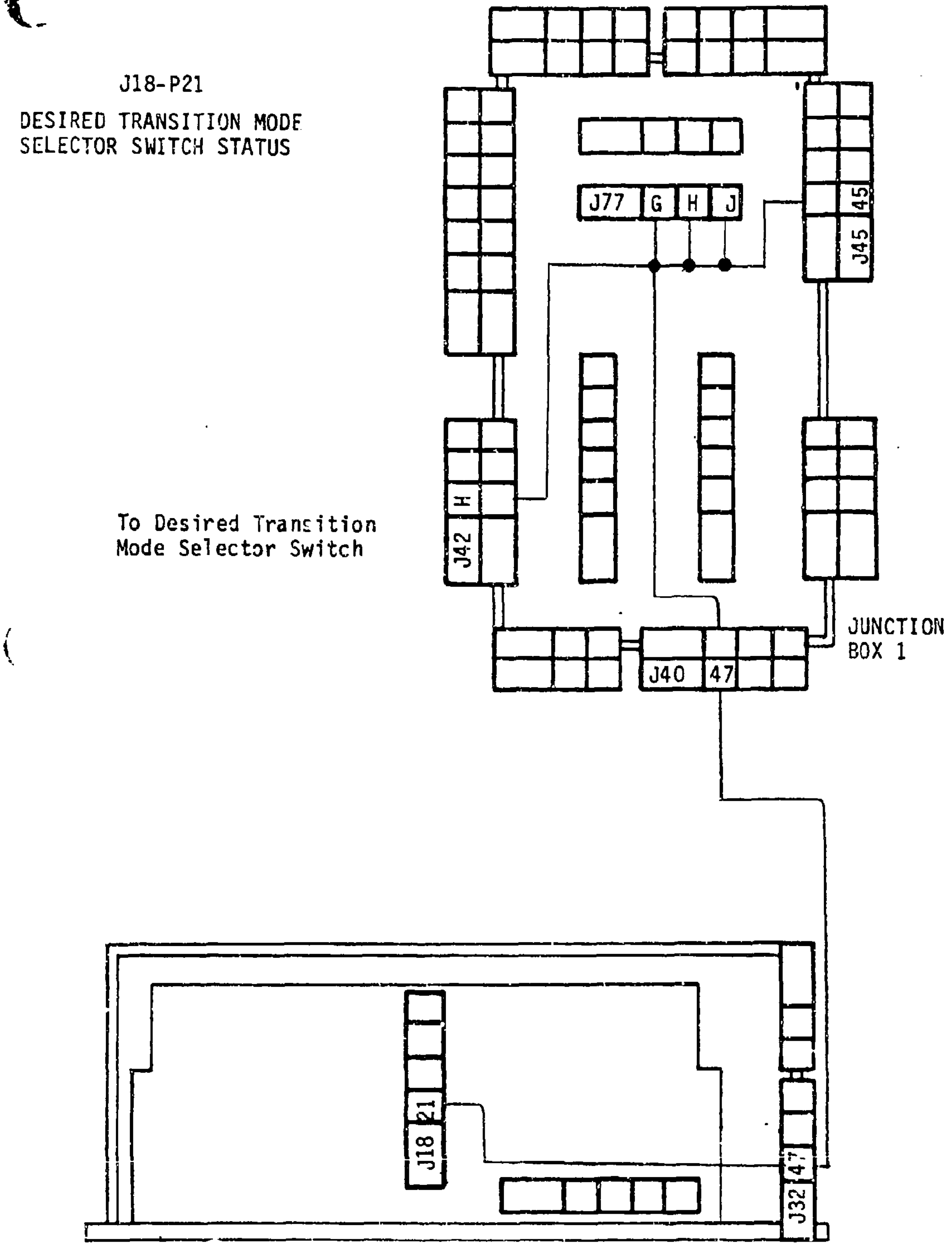
DESIRED LANDBORNE MODE SELECTOR SWITCH STATUS



To Desired Landborne Mode Selector Switch Status

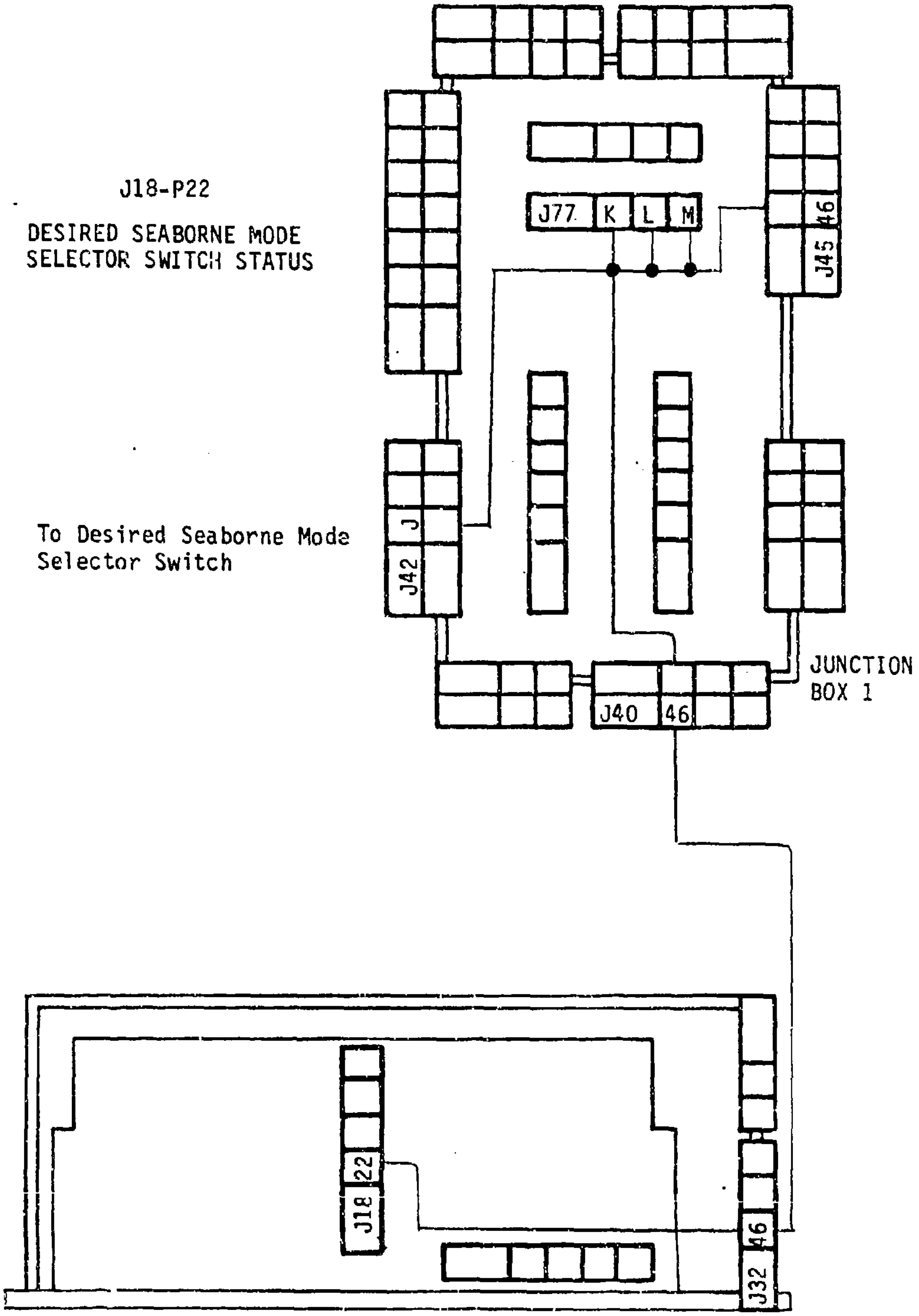
J18-P21

DESIRED TRANSITION MODE SELECTOR SWITCH STATUS



J18-P22

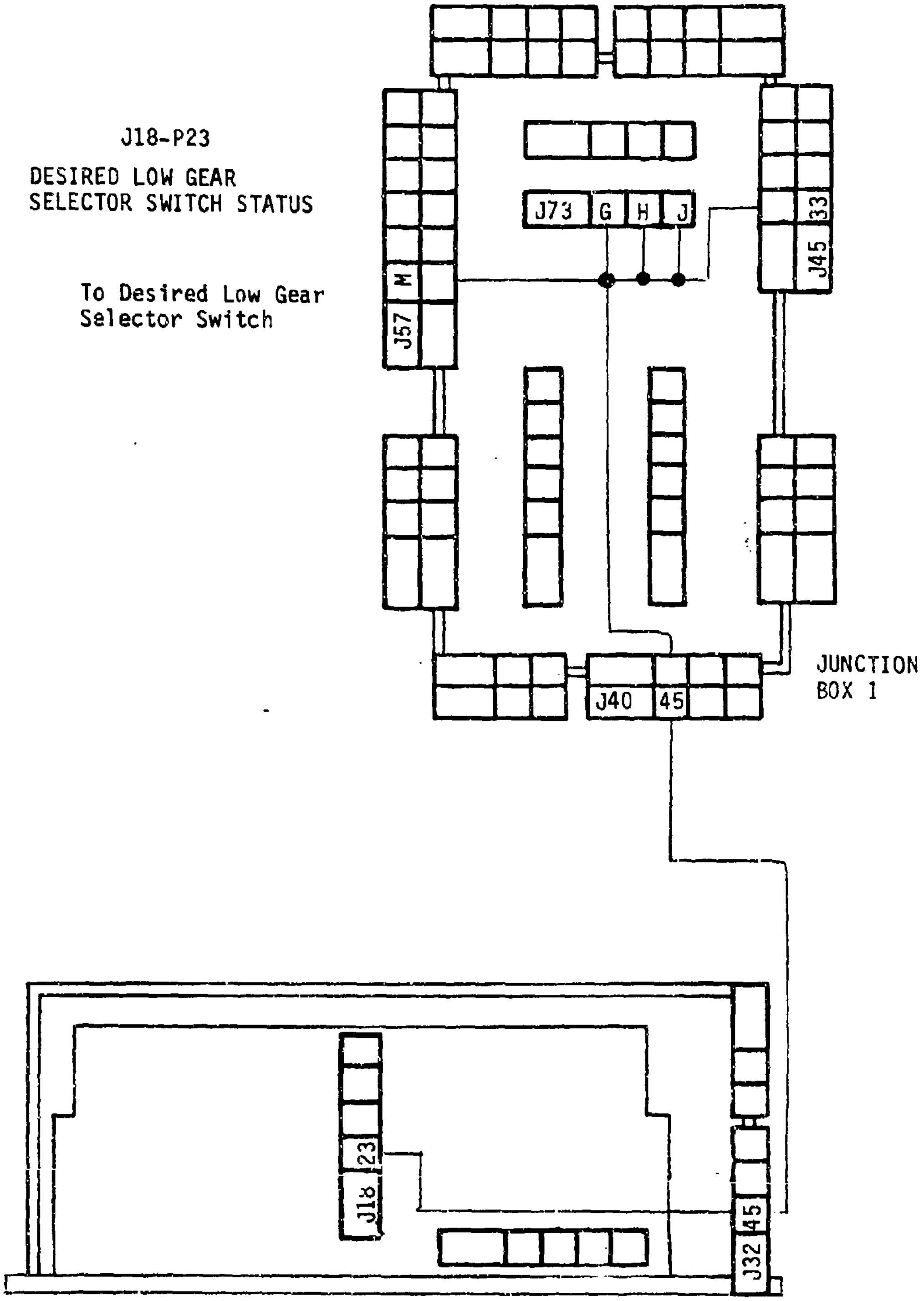
DESIRED SEABORNE MODE SELECTOR SWITCH STATUS



J18-P23

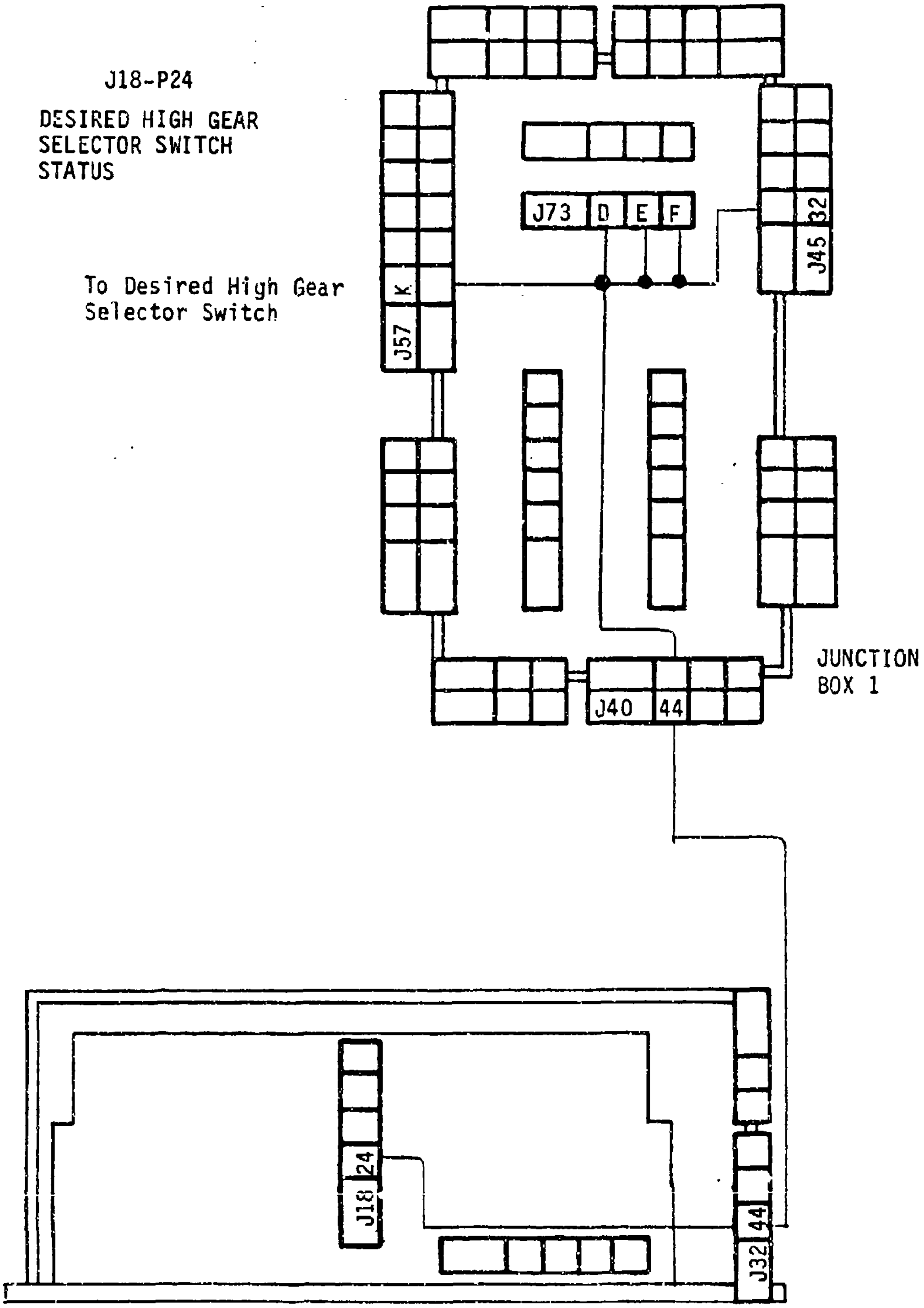
DESIRED LOW GEAR
SELECTOR SWITCH STATUS

To Desired Low Gear
Selector Switch



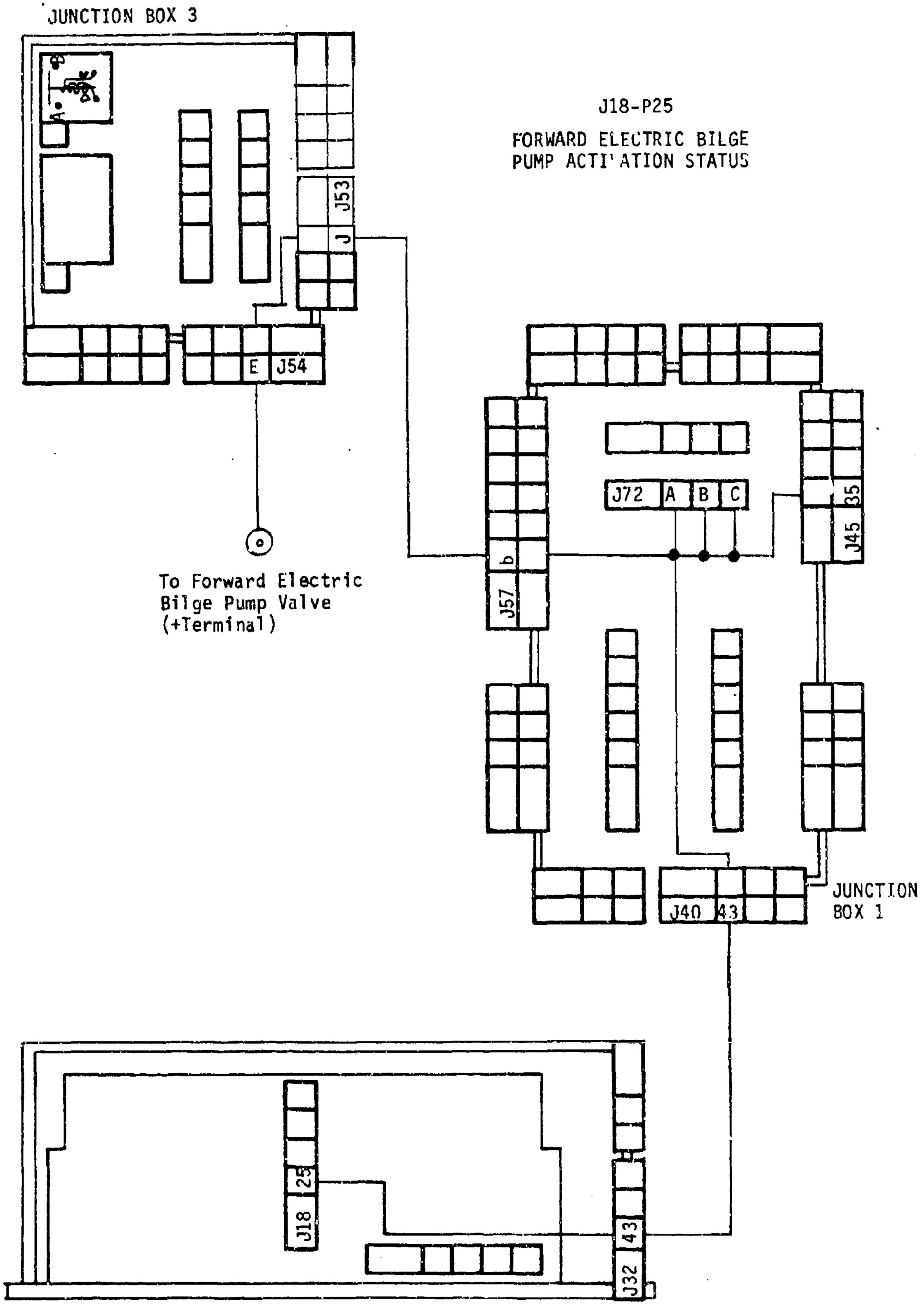
J18-P24

DESIRED HIGH GEAR SELECTOR SWITCH STATUS



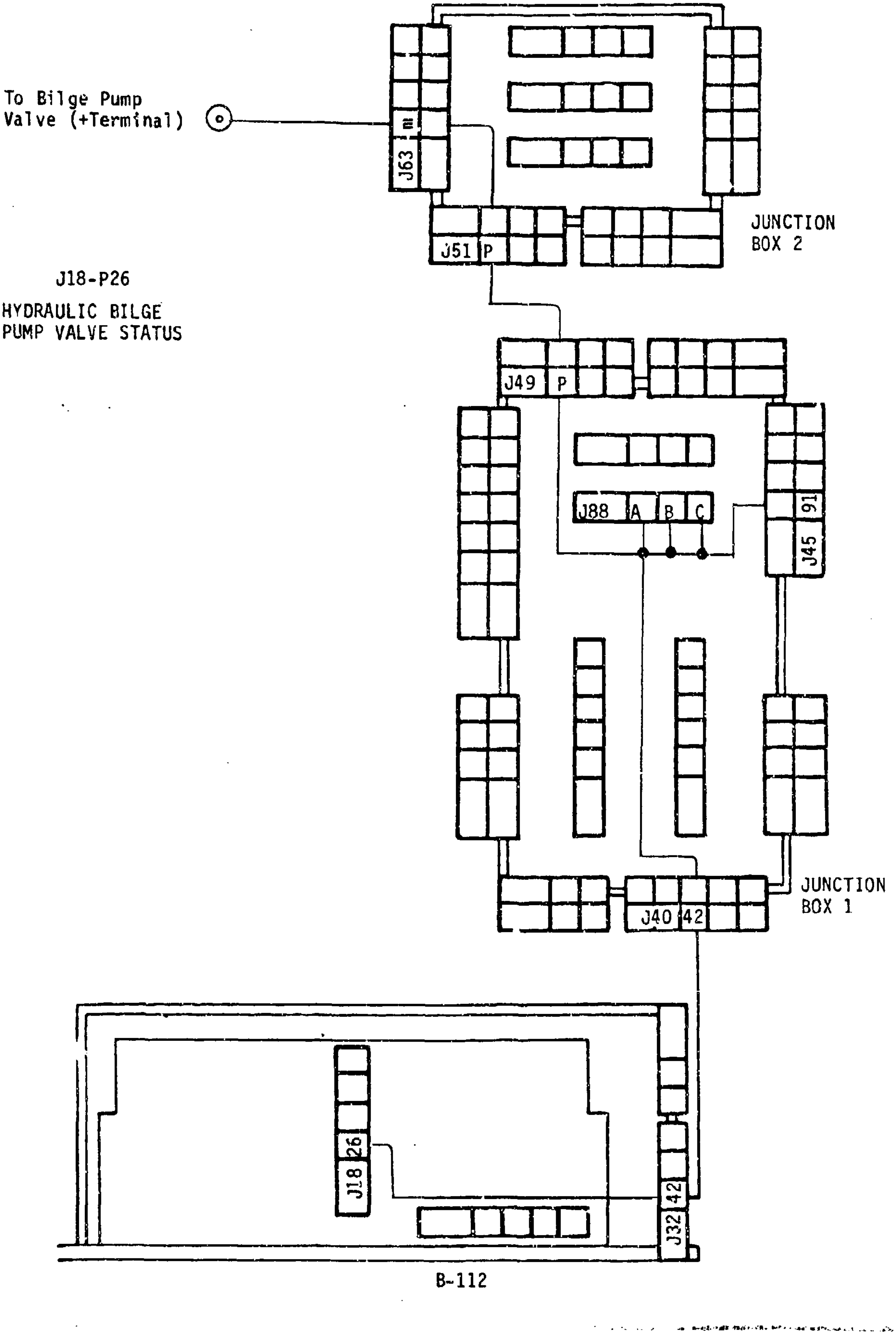
J18-P25

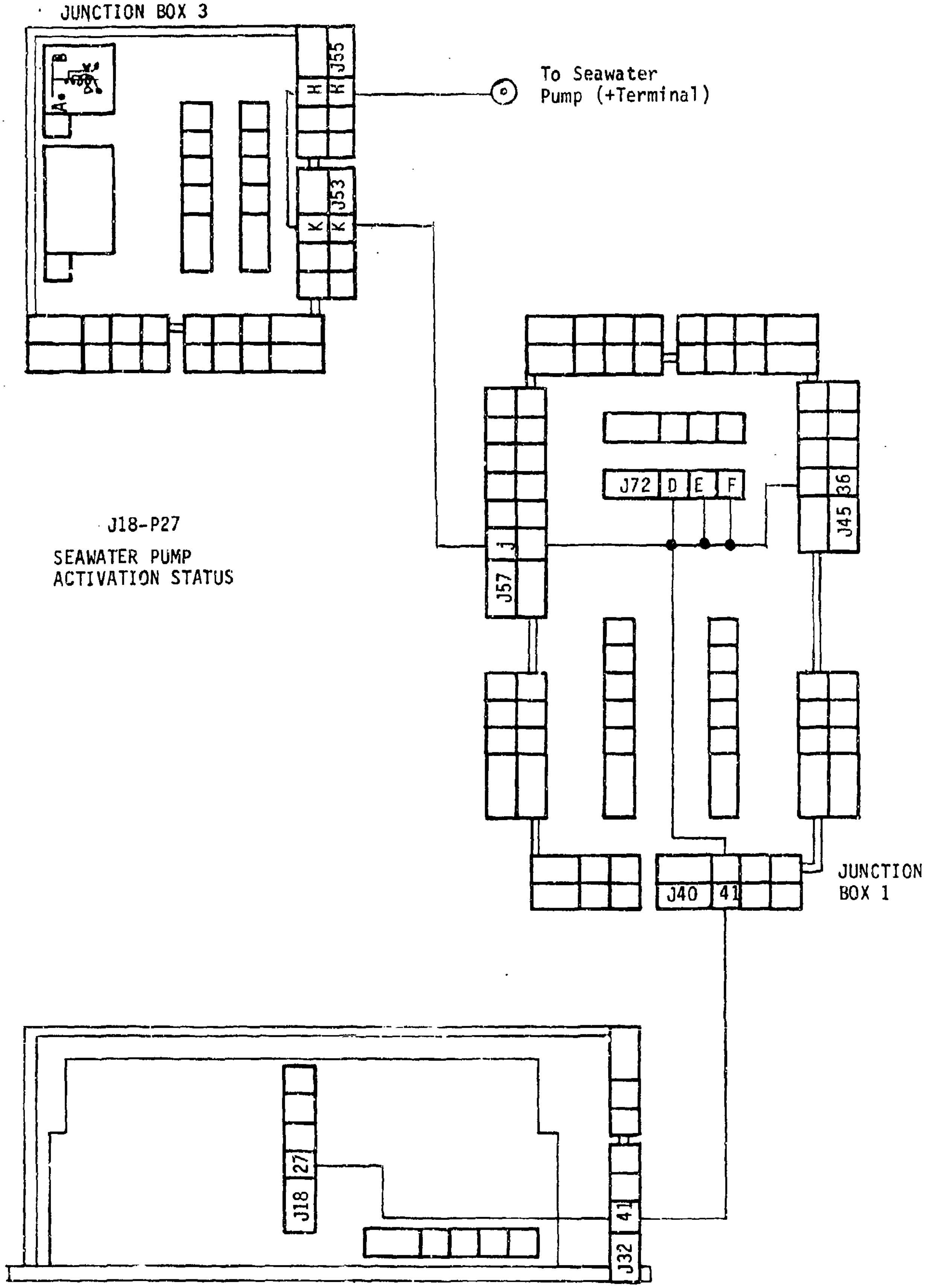
FORWARD ELECTRIC BILGE PUMP ACTI'ATION STATUS



J18-P26

HYDRAULIC BILGE PUMP VALVE STATUS



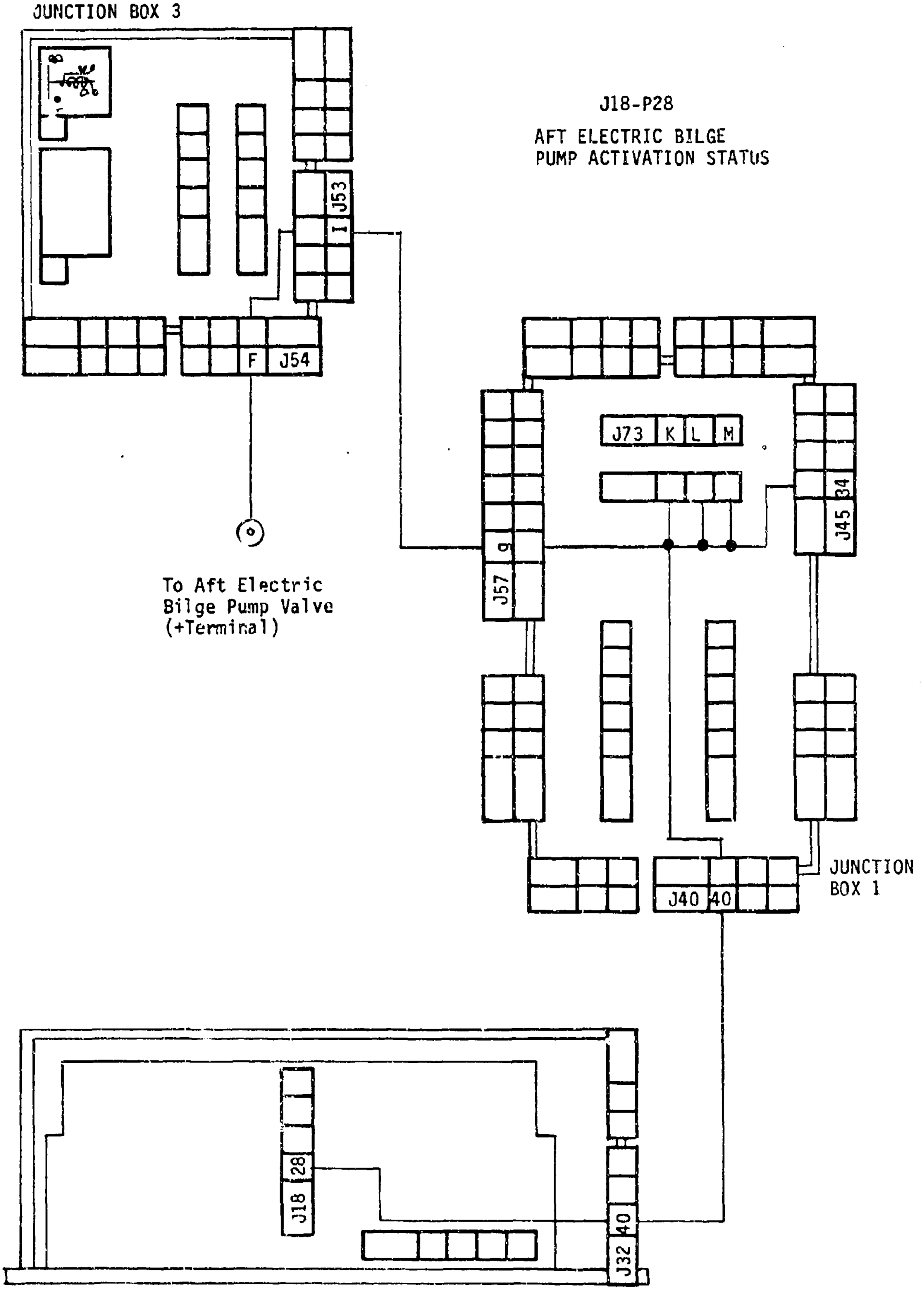


J18-P27

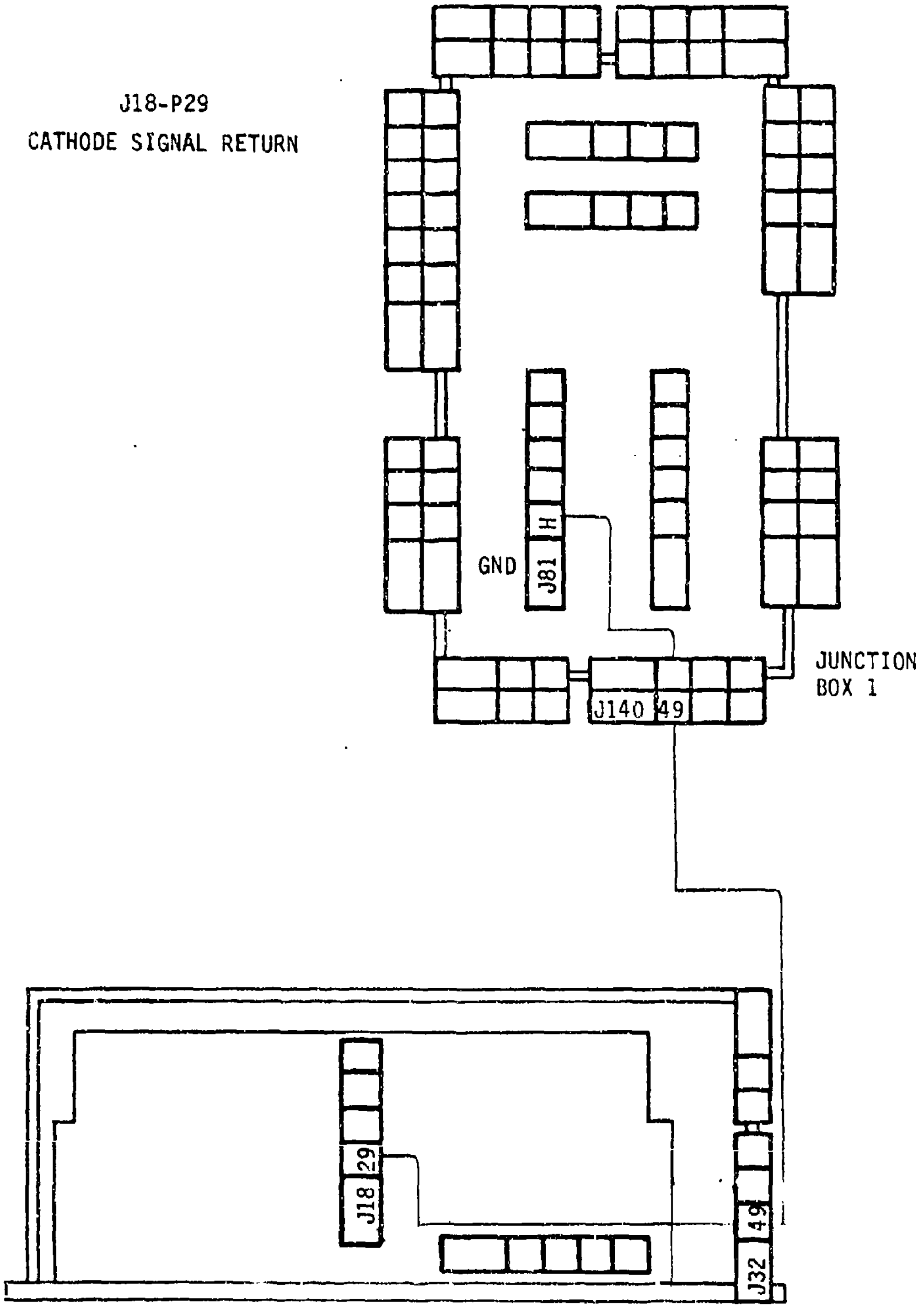
SEAWATER PUMP
ACTIVATION STATUS

J18-P28

AFT ELECTRIC BILGE PUMP ACTIVATION STATUS

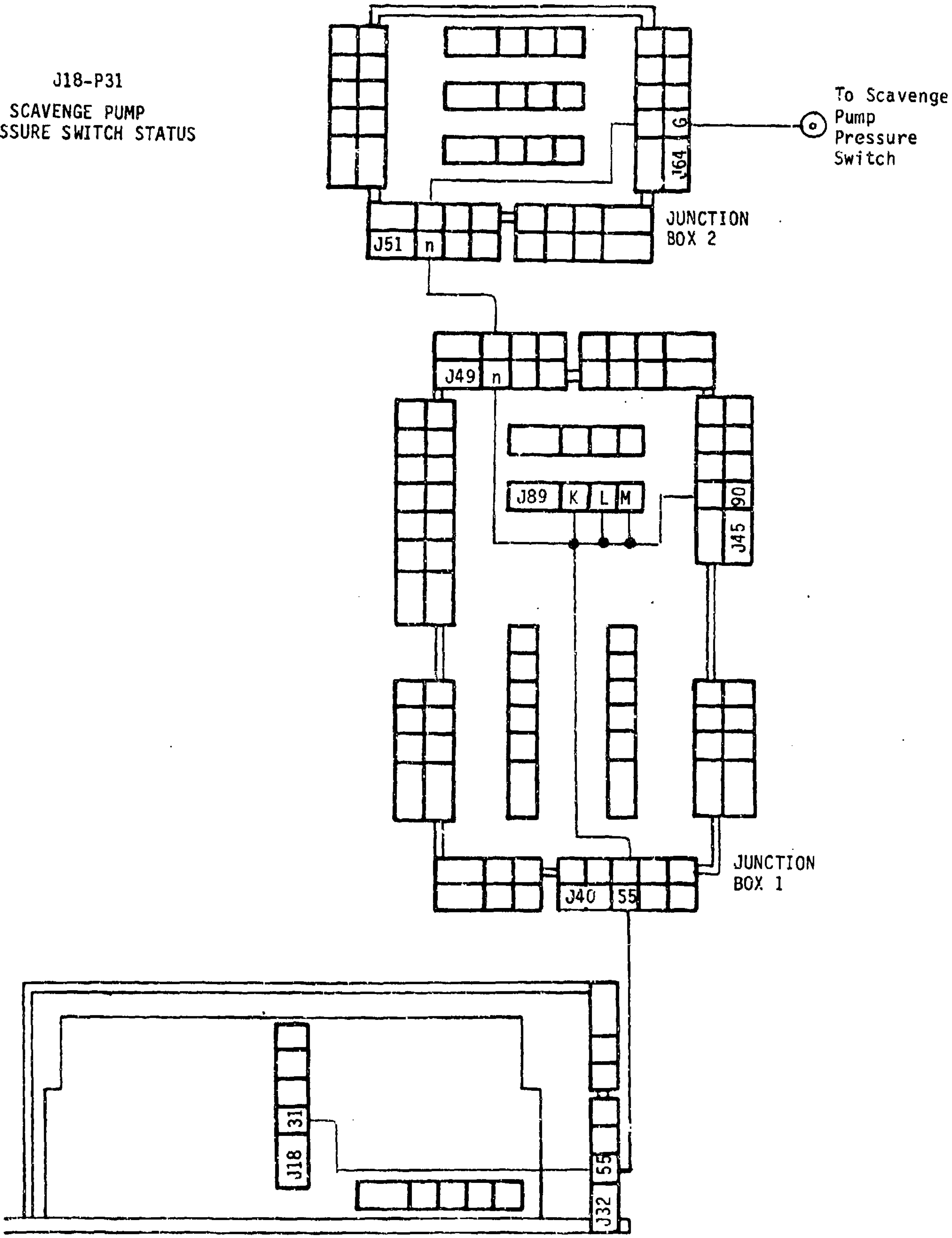


J18-P29
CATHODE SIGNAL RETURN



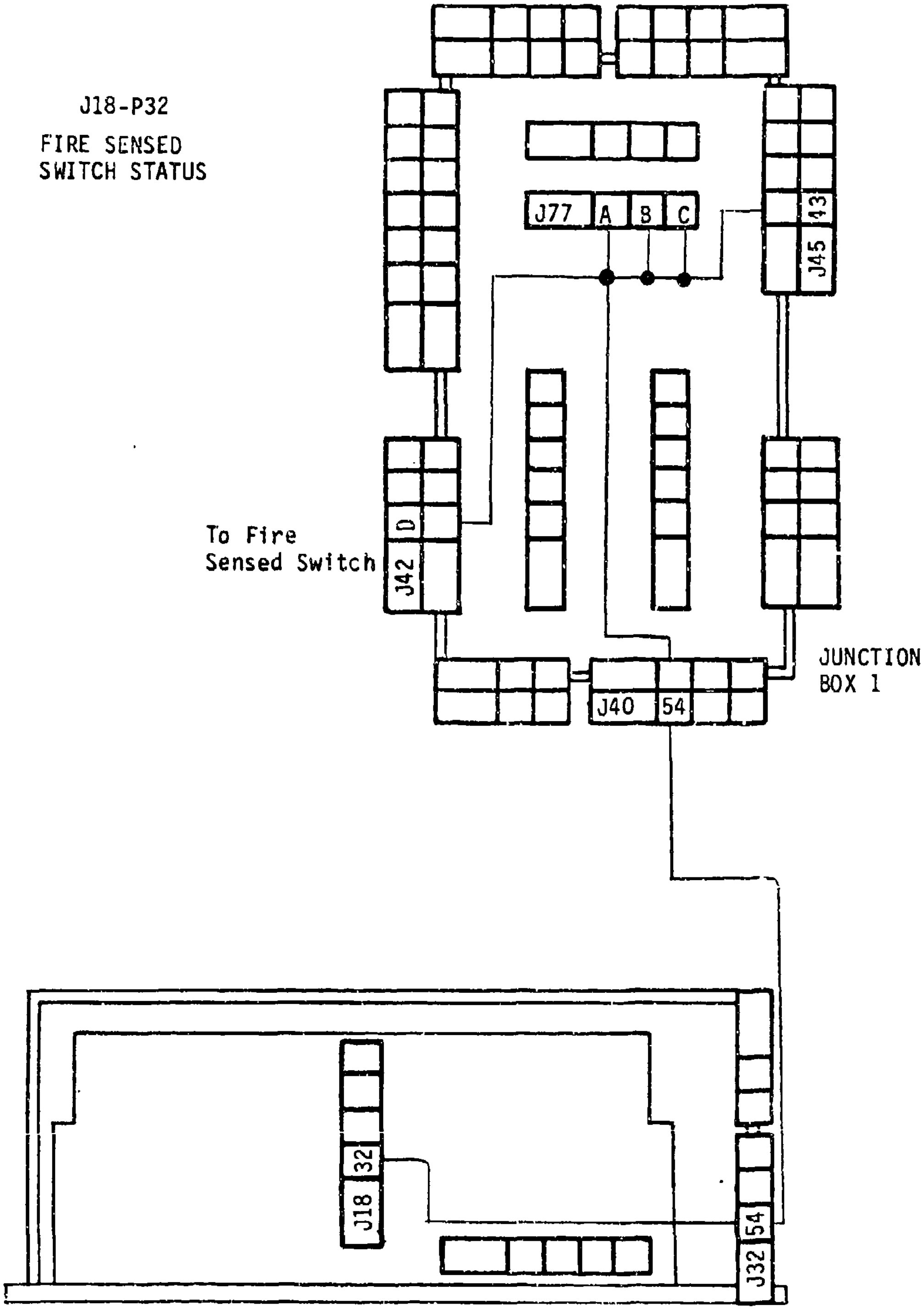
J18-P31

LOW SCAVENGE PUMP
PRESSURE SWITCH STATUS



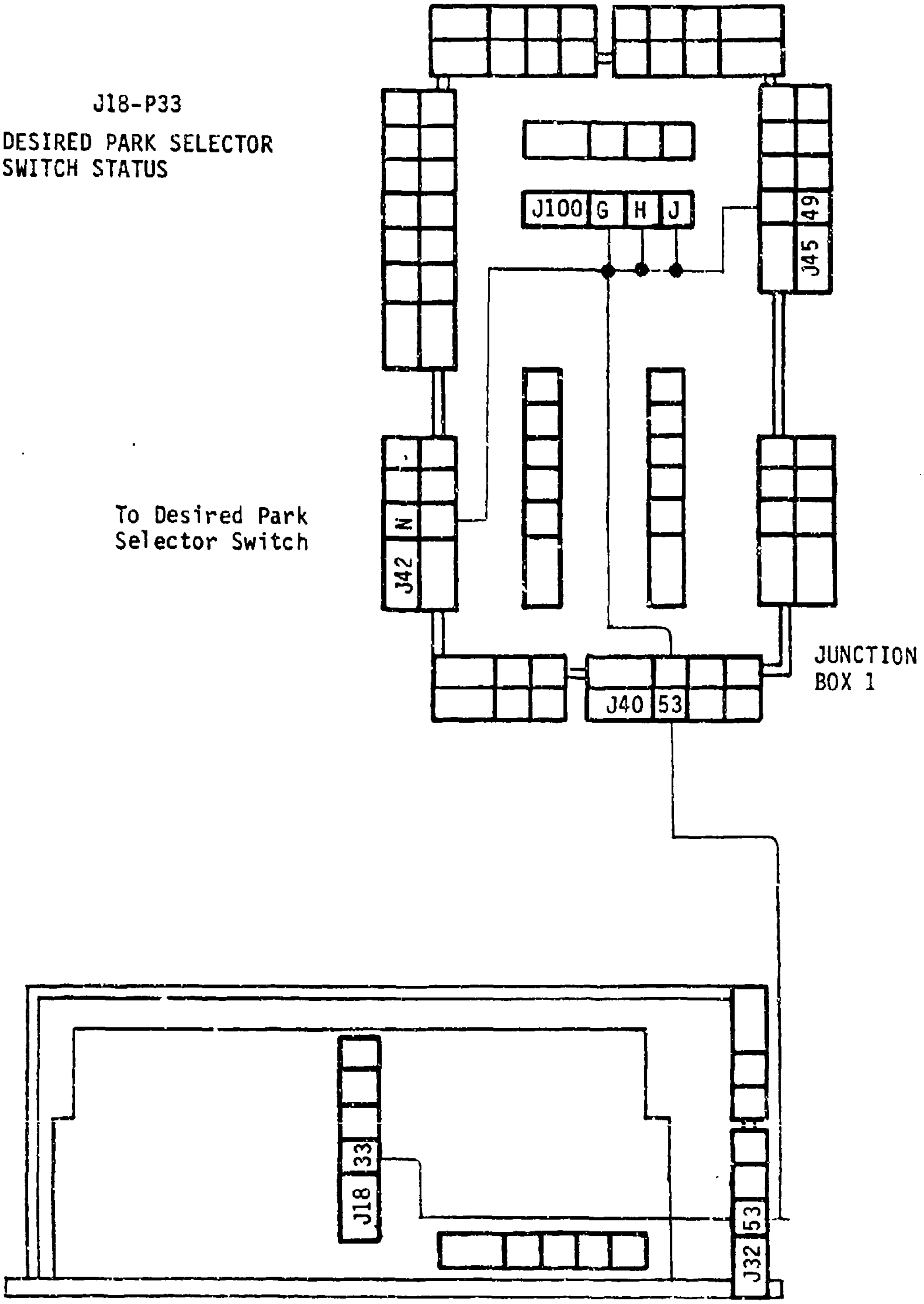
J18-P32

FIRE SENSED
SWITCH STATUS



J18-P33

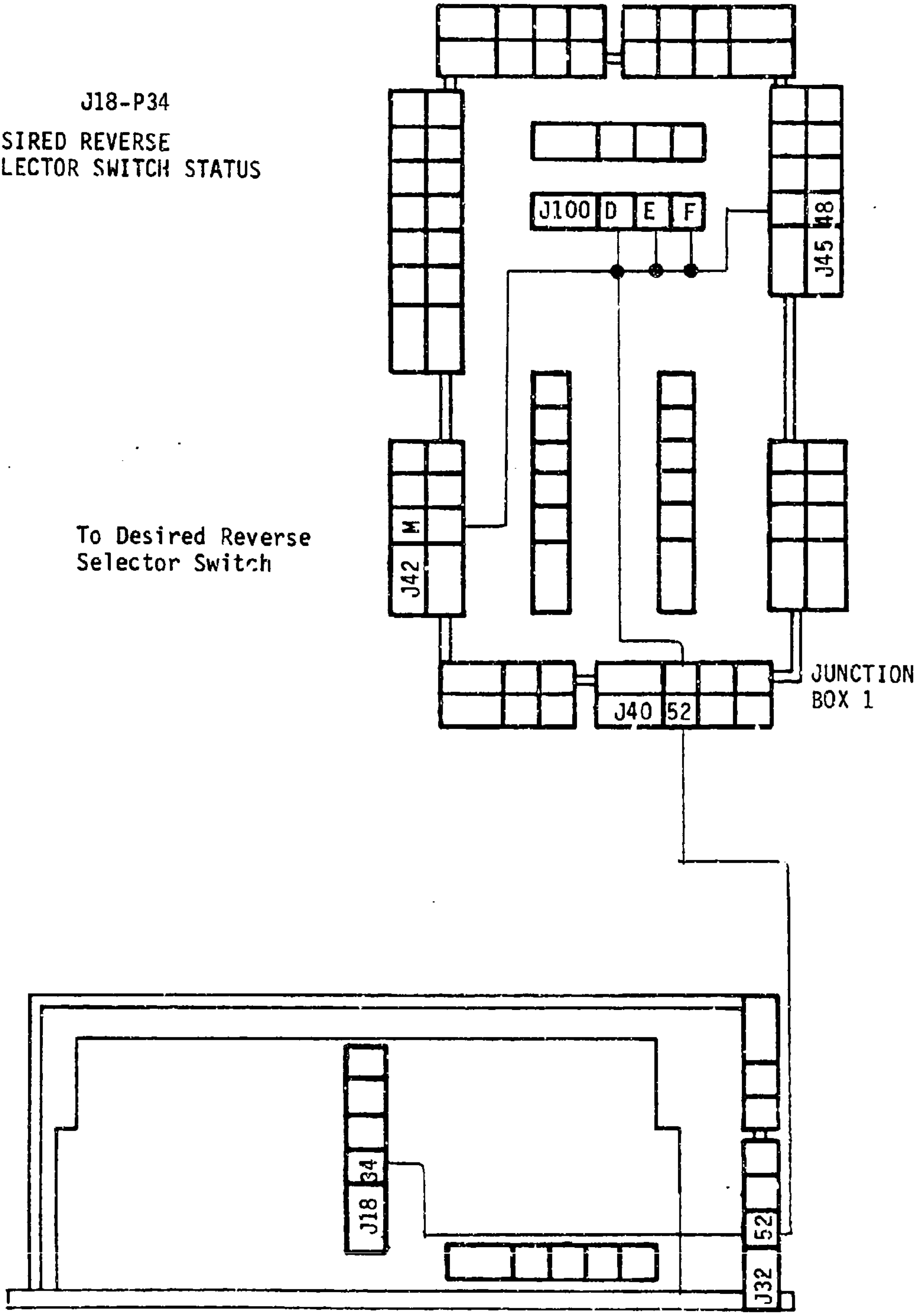
DESIRED PARK SELECTOR SWITCH STATUS



J18-P34

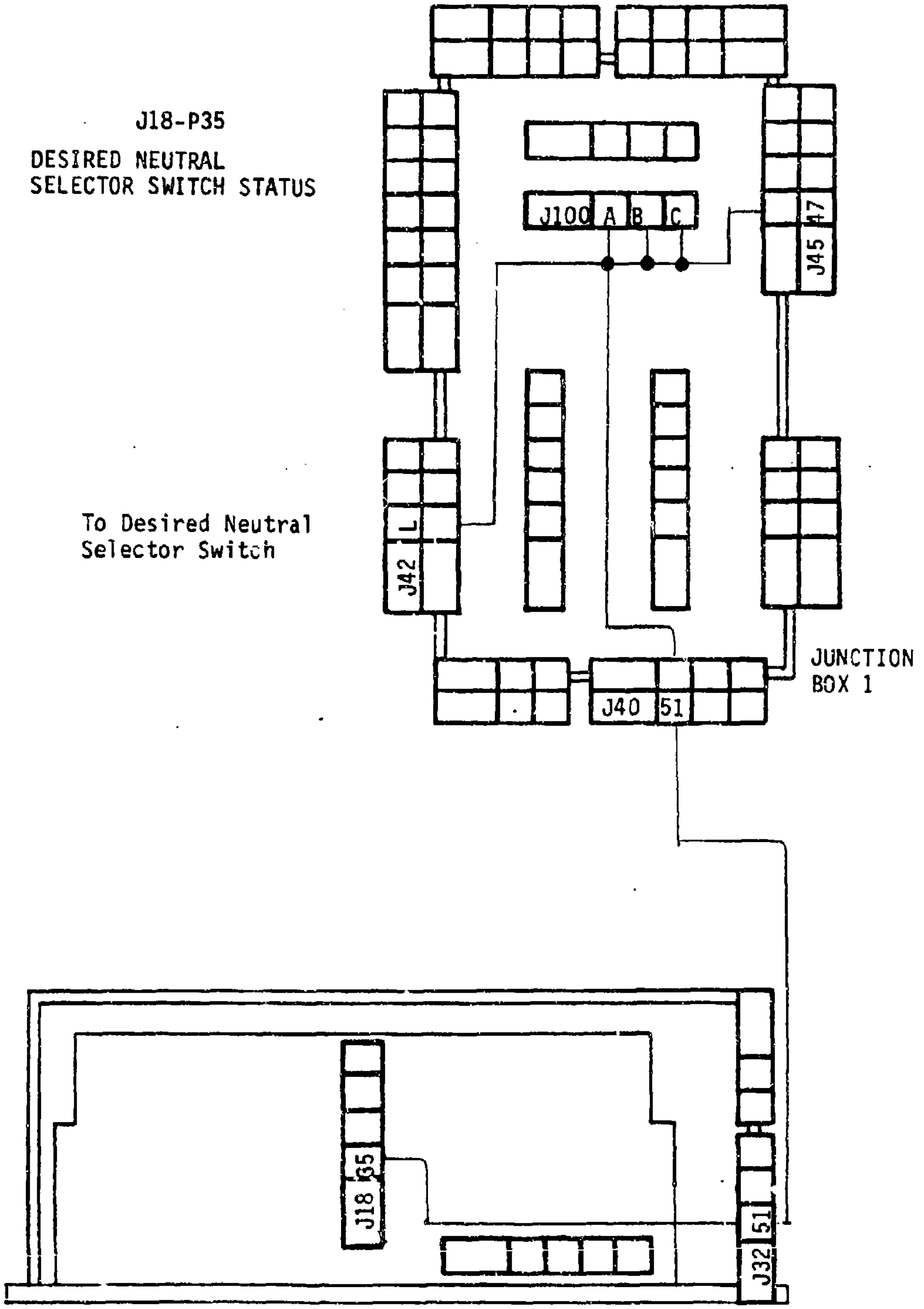
DESIRED REVERSE SELECTOR SWITCH STATUS

To Desired Reverse Selector Switch



J18-P35

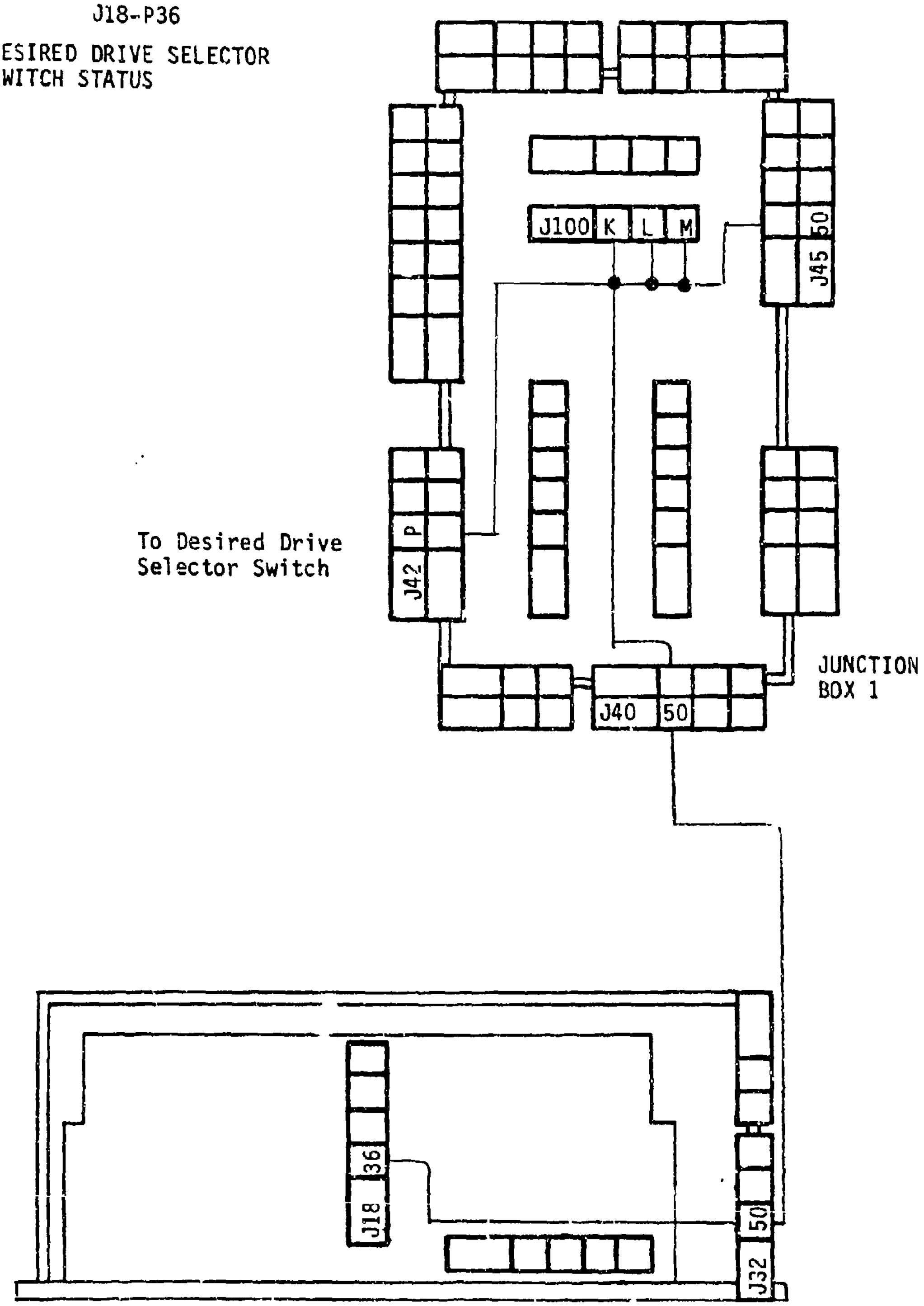
DESIRED NEUTRAL
SELECTOR SWITCH STATUS



To Desired Neutral
Selector Switch

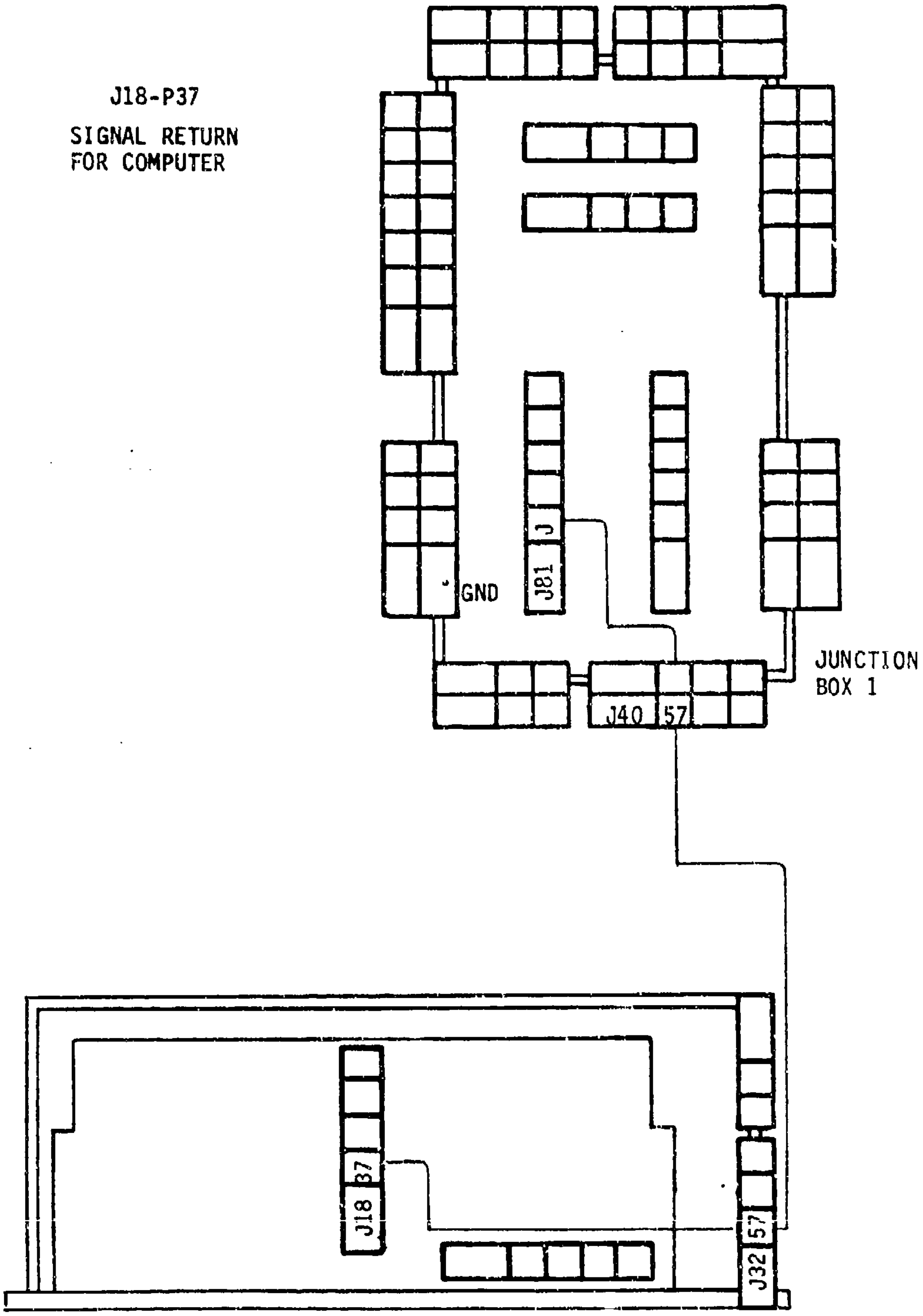
J18-P36

DESIRED DRIVE SELECTOR SWITCH STATUS



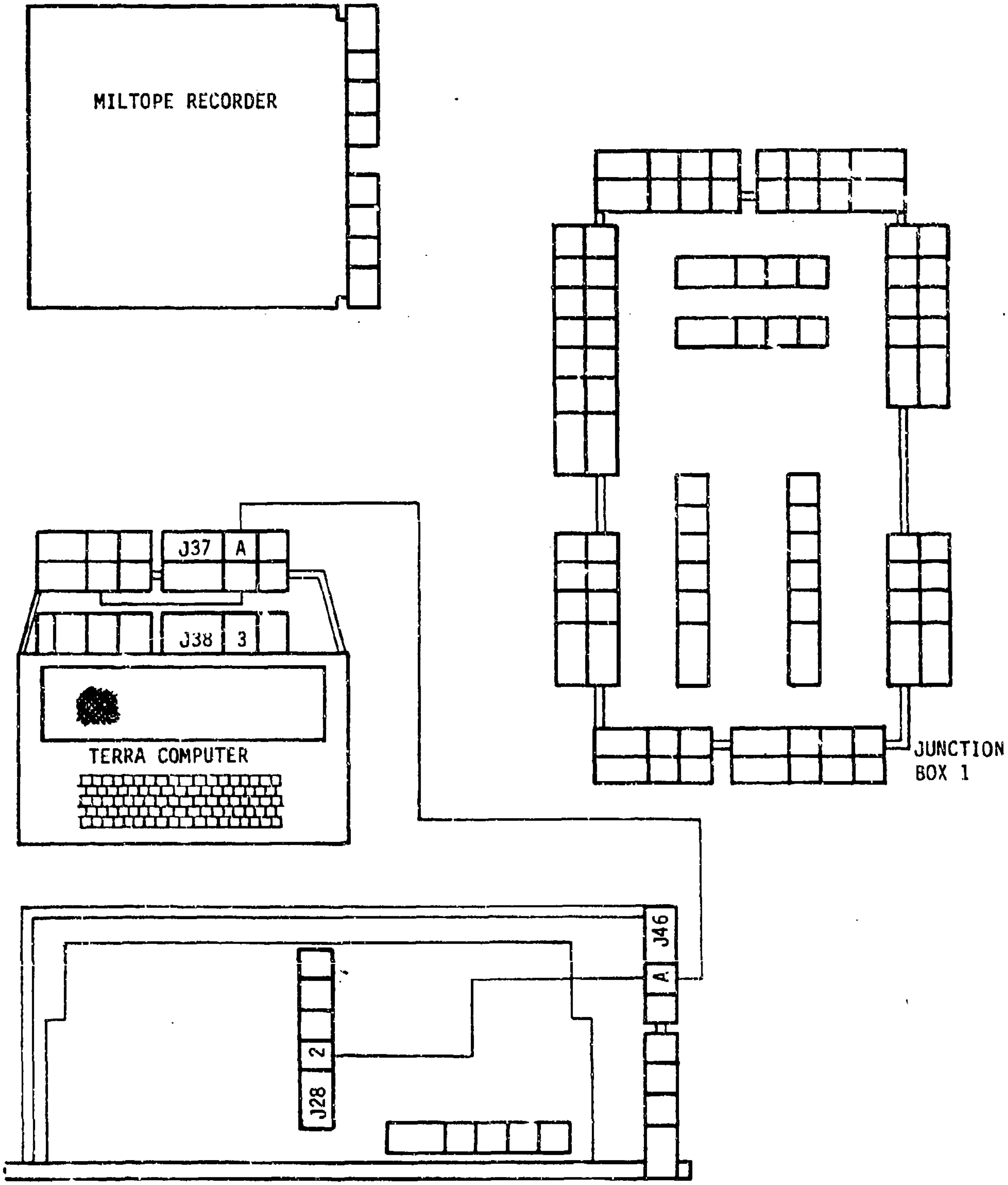
J18-P37

SIGNAL RETURN
FOR COMPUTER



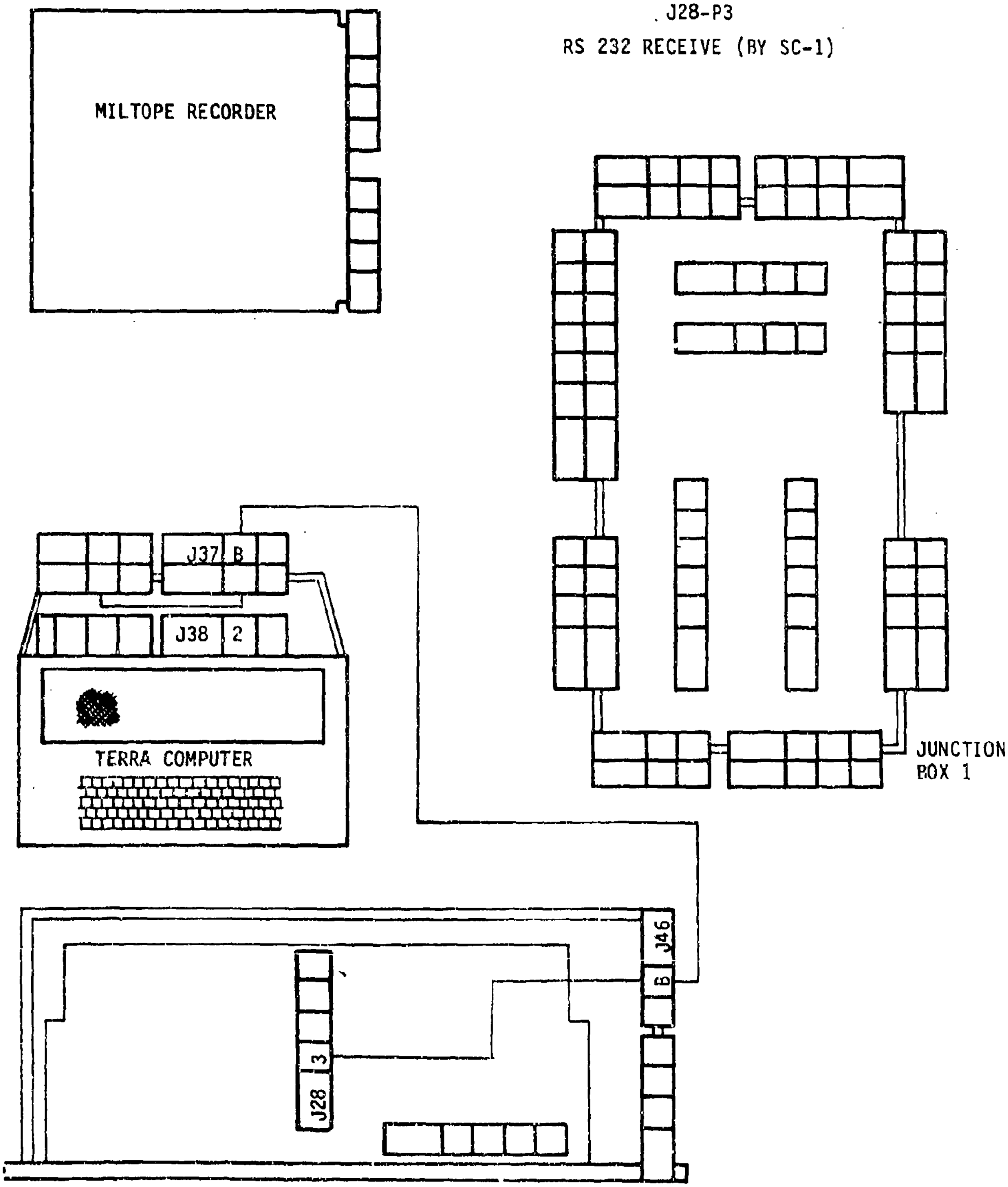
J28-P2

R5232 TRANSMIT (FROM SC-1)



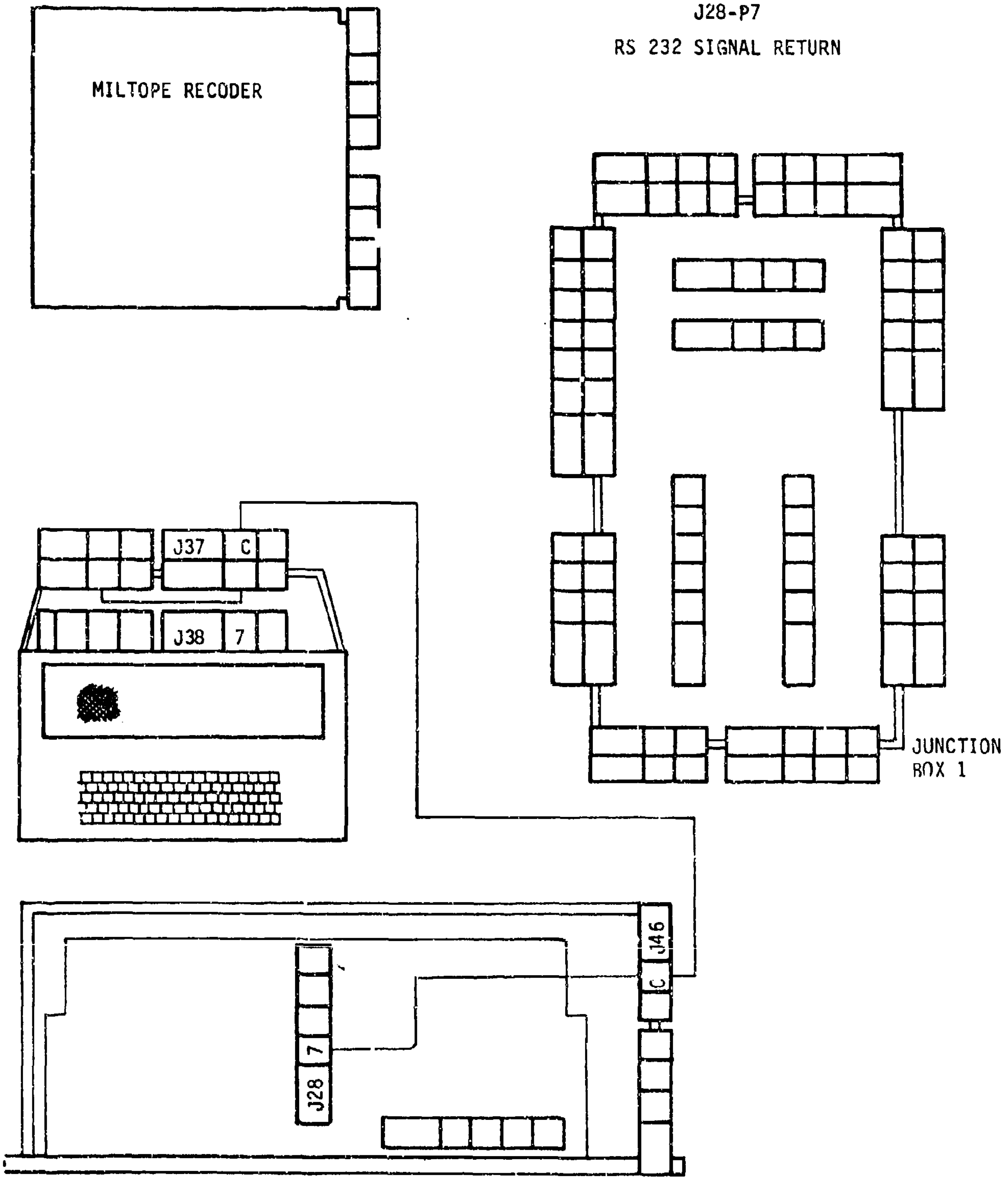
J28-P3

RS 232 RECEIVE (BY SC-1)

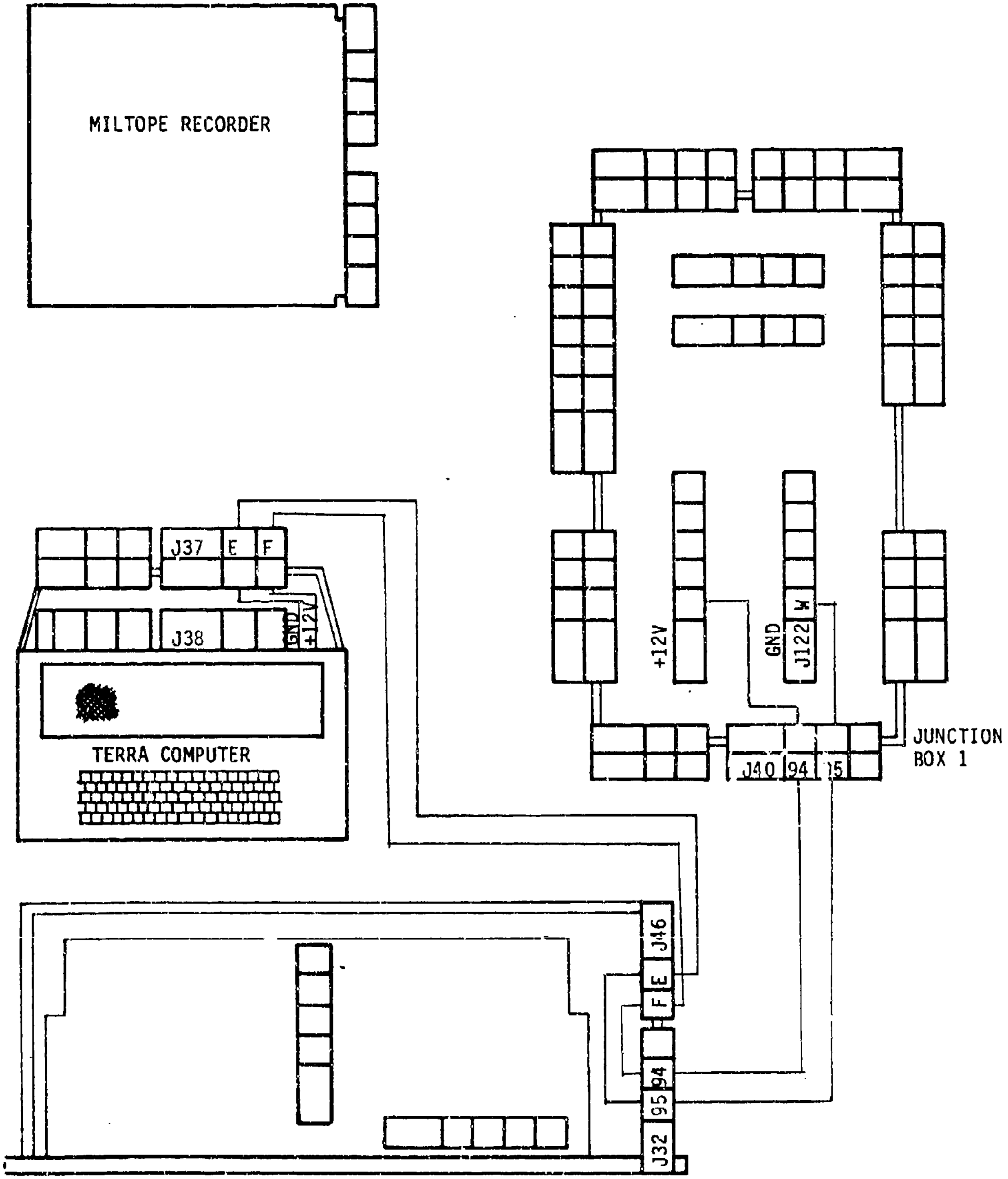


J28-P7

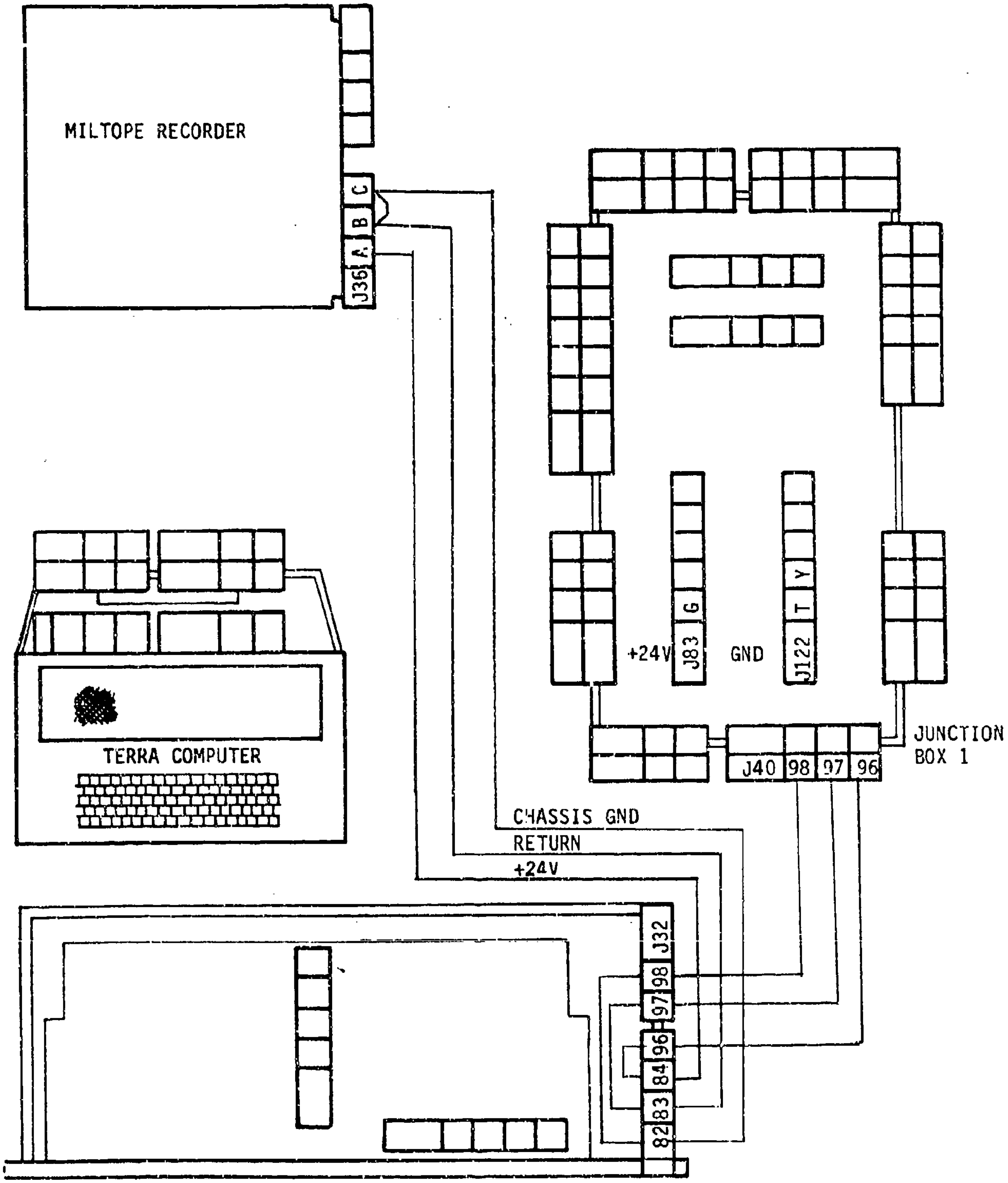
RS 232 SIGNAL RETURN

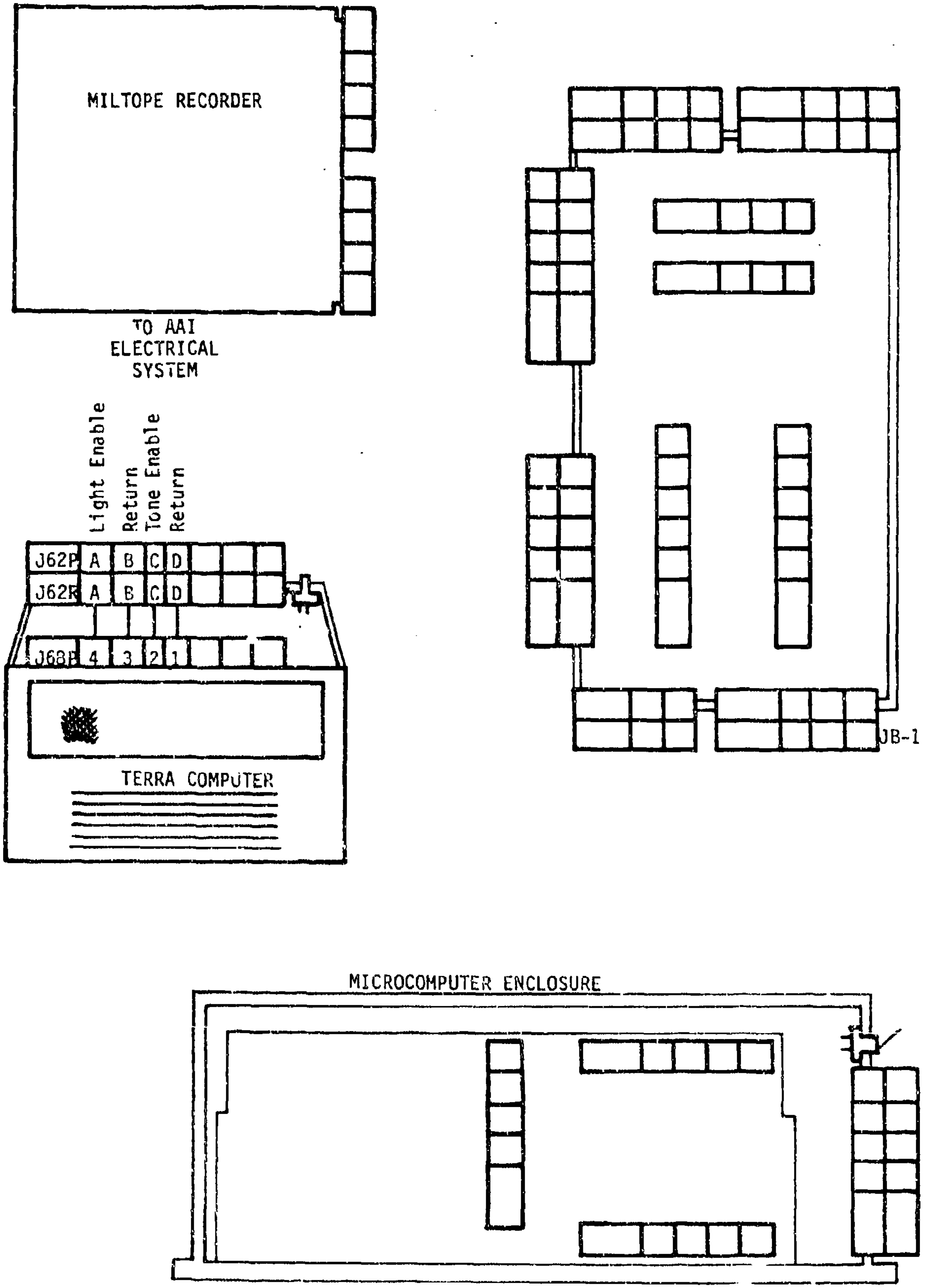


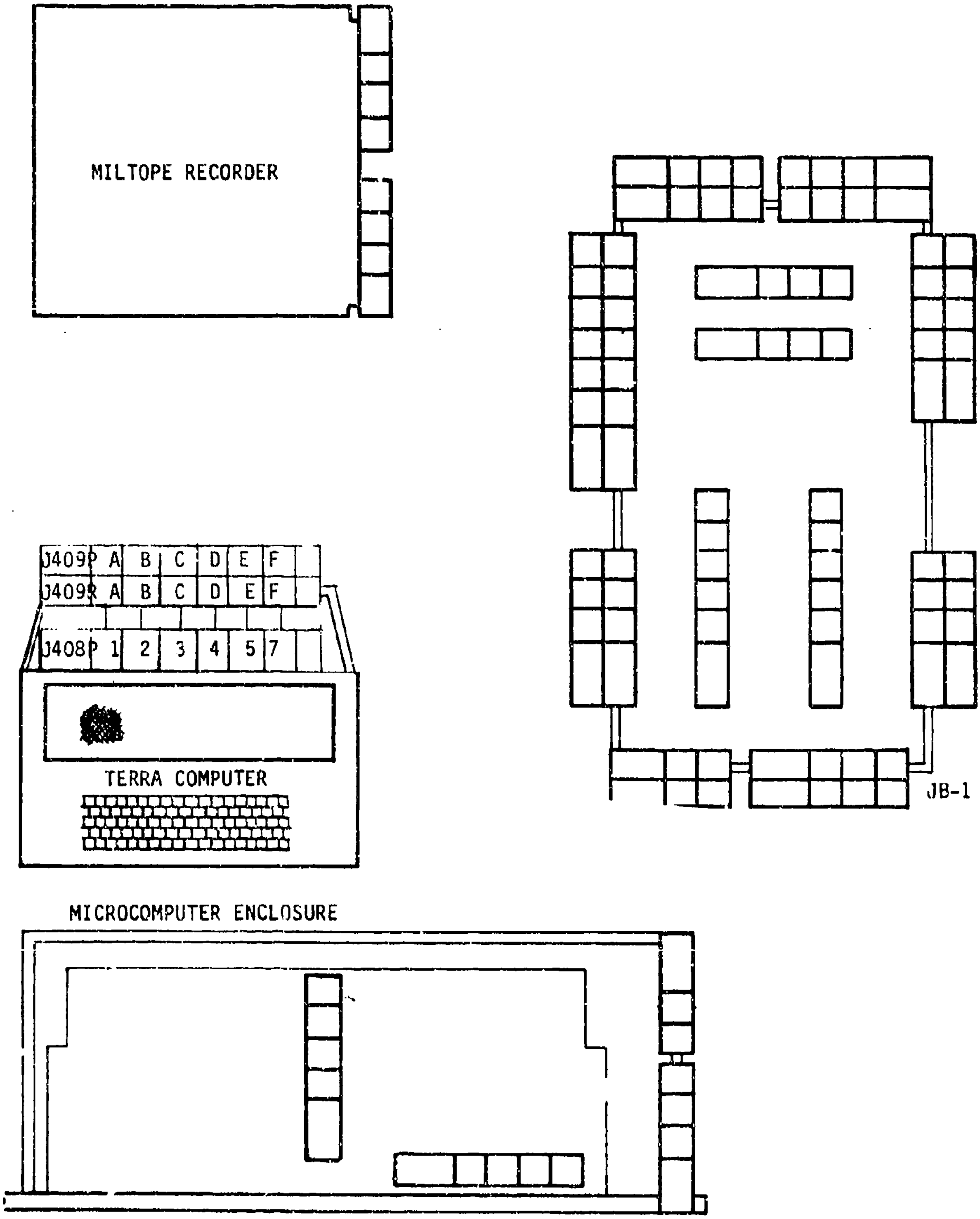
TERRA COMPUTER POWER



MILTOPE POWER AND GROUND CIRCUITS




MILTOPE RECORDER
TO AAI ELECTRICAL SYSTEM
Light Enable
Return
Tone Enable
Return
J62P A B C D
J62R A B C D
J68P 4 3 2 1
TERRA COMPUTER
JB-1
MICROCOMPUTER ENCLOSURE


MILTOPE RECORDER
J409P A B C D E F
J409R A B C D E F
J408P 1 2 3 4 5 7
TERRA COMPUTER
JB-1
MICROCOMPUTER ENCLOSURE